iRobot®

# *iRobot® 1KA Seaglider™ User's Guide*

**Part Number: 4220126**
**Revision: C**
**January 2012**

## Copyright and Liability Information

Copyright ©2012 iRobot Corporation. All Rights Reserved.

Adapted with permission from original material from the University of Washington. Copyright iRobot 2011, rights reserved under the copyright laws of the United States. Any reproduction in part or as a whole without the written permission of iRobot Corporation is prohibited. Seaglider is a trademark of the University of Washington.

**COMMERCIAL ITEM**

The technical data and computer software contained/transmitted herein is adapted with permission from original material from the University of Washington and is proprietary to iRobot Corporation, is iRobot Corporation's commercial technical data and commercial computer software and computer software documentation, and may not be used, disclosed, reproduced, modified, performed, or displayed except as permitted by iRobot's standard commercial license for this data or computer software. United States Government license rights are limited to those rights set forth in iRobot's standard commercial license.

No part of this manual may be copied or distributed, transmitted, transcribed, stored in a retrieval system, or translated into any human or computer language, in any form or by any means, electronic, mechanical, magnetic, manual, or otherwise, including photocopying, or disclosed to third parties, without express written permission from iRobot Corporation, 8 Crosby Drive, Bedford, MA 01730 USA.

**EXPORT CONTROL**

The exportation of this product falls under the jurisdiction of the U.S. Commerce Department Bureau of Industry and Security and is subject to the Export Administration Regulations.

WARNING - This document contains technical data whose export may be restricted by the Arms Export Control Act (Title 22, U.S.C., Sec 2751, et seq.) or the Export Administration Act of 1979, as amended (Title 50, U.S.C., App. 2401 et seq.).

VIOLATIONS OF THESE EXPORT LAWS ARE SUBJECT TO SEVERE CRIMINAL PENALTIES.

**Limits of Liability**

Although every precaution has been taken in the preparation of this documentation, iRobot Corporation assumes no responsibility whatsoever for errors or omissions, or for damages resulting from the use of the information contained herein.

To the maximum extent permitted by applicable law, iRobot Corporation, its officers, employees and contractors, and their suppliers disclaim all warranties, either expressed or implied, including, but not limited to, implied warranties of merchantability and fitness for a particular purpose, with regard to the hardware, software, and all accompanying or subsequently supplied written materials and documentation.

To the maximum extent permitted by applicable law, in no event shall iRobot Corporation, its officers, employees or contractors, or their suppliers, be liable for any damages whatsoever (including without

limitation, special, incidental, consequential, or indirect damages for personal injury, loss of business profits, or business interruption).

**Warranty**

This product comes with a one year limited warranty. See Appendix J, “Warranty and Disclaimers” on page 441. The pressure hull of the iRobot® 1KA Seaglider™ does not contain any user serviceable items. Any unauthorized opening of the pressure hull by customers voids this one year warranty.

**Special Note**

Because some states/jurisdictions do not allow the exclusion or limitation of liability for consequential or incidental damages, the above limitations may not apply to you. The warranty gives you specific legal rights, and you may also have other rights, which vary from state/jurisdiction to state/jurisdiction.

**Trademarks**

iRobot and PackBot are registered trademarks and SUGV is a trademark of iRobot Corporation.

Iridium and the Iridium logo are registered trademarks and/or service marks of Iridium Satellite LLC.

MAC, Mac, and Mac OS are trademarks of Apple Inc., registered in the U.S. and other countries.

PAINE ® is a registered trademark of Paine Electronics, LLC

PicoDos® is Persistor Instrument’s DOS-like operating system for the CF8/TT8 combination used on iRobot® 1KA Seaglider™. It provides access to the DOS FAT file system on the Compact Flash, as well as some simple file manipulation utilities. The TOM8 and PicoDOS® commands are documented in the PicoDOS® User's Guide of November, 2000 (which is incorporated into this document by reference). The extensions below provide additional functionality, either to extend PicoDOS® generally or to provide 1KA Seaglider™-specific functions.

Tera Term (Pro) is a free software terminal emulator (communication program) for MS-Windows. It supports VT100 emulation, telnet connection, serial port connection, and so on.

Windows, Windows 95, Windows 98, Windows Me, Windows NT, Windows 2000, Dial-Up Networking, Outlook, Outlook Express and Internet Explorer are registered trademarks of Microsoft Corporation.

Other product and company names mentioned may be trademarks or registered trademarks of their respective companies. Mention of third-party products is for informational purposes only and constitutes neither a recommendation nor an endorsement.

# *Table of Contents*


**Preface** ..... *17*

Who Should Use this Manual ..... **17**
Typographical Conventions ..... **18**
Safety Conventions ..... **19**
Safety Information ..... **19**
*Personal Safety Information* ..... ***19***
*Vehicle Hazards* ..... ***20***
*Battery Safety* ..... ***20***
Registration ..... **20**
iRobot Contact Information ..... **20**
*Customer Support* ..... ***20***
*Sales* ..... ***21***

**CHAPTER 1** *1KA Seaglider* ..... ***23***

Seaglider History ..... **23**
System Overview ..... **25**
Seaglider Sensors ..... **27**
*Standard Seaglider Sensors* ..... ***27***
*Optional Sensors* ..... ***28***

**CHAPTER 2** *Checking Out Crate Contents* ..... ***29***

Inspecting Seaglider's Shipping Container ..... **30**
Inspecting the Basestation Container ..... **31**

**CHAPTER 3** *Setting Up the System* ..... ***33***

Setting Up the Basestation ..... **34**
*Basestation Configuration* ..... ***34***
*Basestation Log In* ..... ***34***
*Internet Configuration* ..... ***35***
*Setting Up the Basestation Directories and Files* ..... ***35***
Setting Up Seaglider's Interface Computer ..... **37**

Setting Up the Pre-Flight Diagnostic Laptop ..... 37
Assembling Seaglider's Cradle ..... 38
Assembling Seaglider ..... 41
Stowing/Moving Seaglider ..... 46
Checking Out the Seaglider System ..... 47
*Checking Seaglider Communication with the Field Laptop ..... 47*
*Checking the Primary and Alternate Phone Numbers ..... 52*
*Checking the Pump, Motors, and Basestation Communication ..... 54*

**CHAPTER 4** *Operating Principles ..... 59*

Seaglider Components ..... 60
*Isopycnal Pressure Hull ..... 60*
*Inside the Pressure Hull ..... 61*
*Outside the Pressure Hull ..... 63*
*Inside and Outside the Pressure Hull ..... 65*
Principles of Seaglider Operation ..... 65
*Environmental Factors ..... 65*
*Static Forces ..... 66*
*Gravity ..... 67*
*Buoyancy ..... 67*
*Dynamic Forces ..... 68*
*Hydrodynamic Model ..... 69*
Control of the Static Forces ..... 69
*Pitch ..... 70*
*Roll ..... 71*
*Buoyancy ..... 72*
Features of Control ..... 73
*Canonical Dive ..... 73*
*Control Design ..... 74*
*Sampling ..... 76*
*Sensor Interfaces ..... 76*
*Bathymetry ..... 76*
*Run Phases ..... 77*
*Launch ..... 78*
*Surface ..... 78*
*Dive Phase ..... 80*
*Apogee ..... 80*
*Climb ..... 81*
*Recovery ..... 81*

**CHAPTER 5** *Piloting Parameters* ..... ***83***

Parameter Conventions ..... **83**
*Command File (cmdfile) State Directives* ..... ***84***
Piloting Parameters ..... **85**
Parameters by Category ..... **86**
*Most-Frequently Modified Parameters* ..... ***86***
*Second Most-Frequently Modified Parameters* ..... ***86***
*Less Frequently-Modified Parameters* ..... ***87***
*Parameters Set During Fabrication* ..... ***89***
*Alphabetized Parameters* ..... ***92***

**CHAPTER 6** *Pre-Deployment Tasks* ..... ***159***

Mission Planning ..... **160**
*Environment* ..... ***160***
*Endurance* ..... ***161***
*cmdfile Edits* ..... ***166***
*targets Edits* ..... ***167***
*Science Sensors* ..... ***168***
*science Edits* ..... ***169***
*Bathymetric Files* ..... ***171***
*.pagers file* ..... ***172***
*.mailer file* ..... ***172***
*.URL file* ..... ***173***
Transporting Seaglider to the Field ..... **174**
*Detaching the Antenna Mast* ..... ***174***
*Putting Seaglider in Travel Mode* ..... ***175***
*Packing Seaglider in its Shipping Case* ..... ***177***
*Shipping Seaglider* ..... ***178***
*Assembling Seaglider* ..... ***178***
*Performing the Interactive Self Test* ..... ***178***
*Communication Messages* ..... ***179***
Final Launch Procedure ..... **180**
*ARGOS tag* ..... ***186***

**CHAPTER 7** *Pre-Launch Procedures* ..... ***187***

Interactive Pre-launch Self Test ..... **188**
Simulated Dives ..... **219**
*ARGOS tag* ..... ***227***

**CHAPTER 8** *Tuning Your Seaglider* ..... ***229***

Tuning Your Seaglider ..... **229**

**CHAPTER 9** *Files for Operations* ..... ***237***

Files Placed on the Basestation by the Pilot ..... **237**
*Command File* ..... ***239***
*Targets File* ..... ***239***
*ALI Sensors* ..... ***242***
*Science File* ..... ***243***
*PicoDOS Commands Batch File* ..... ***245***
Factory-Generated File ..... **246**
*Calibration File* ..... ***246***
Seaglider-Generated Files ..... **246**
*Capture Files* ..... ***246***
*Data Files* ..... ***247***
*Log Files* ..... ***247***
*Files Generated by the GPCTD* ..... ***248***
Basestation-Generated Files ..... **248**
*ASCII Files (.asc)* ..... ***248***
*Baselog File* ..... ***249***
*Binned Profiles Files* ..... ***249***
*Communications File* ..... ***249***
*Engineering Files* ..... ***249***
*netCDF Files* ..... ***250***
*Pagers File* ..... ***250***
*.pro Files* ..... ***250***
*Processed Files Cache File* ..... ***251***
*Private Files* ..... ***251***
*URLs File* ..... ***251***
*Mailers* ..... ***251***

Files Stored on Seaglider ..... 253
*Bathymetry Map Files* ..... *253*
*Battery File* ..... *253*
*Capvec File* ..... *253*
*Compass Calibration File* ..... *254*

**CHAPTER 10** *Recovery and Disassembly* ..... *255*

Recovery Loop ..... 255
Recovery Phase ..... 256

**CHAPTER 11** *Operator Level Maintenance* ..... *259*

Cleaning Seaglider ..... 259
Deep Cleaning Seaglider ..... 261

**CHAPTER 12** *1KA Seaglider Refurbishment* ..... *263*

iRobot Refurbishment ..... 263
Contact Information ..... 265
Service Information ..... 265
Standard Services ..... 266
Upgrade Services ..... 267
Replacement Spares, Sensors & Accessories ..... 268
Diagnostics ..... 269

**APPENDIX A** *System Specifications* ..... *271*

**APPENDIX B** *Seaglider File Formats Manual* ..... *275*

Chapter 1 ..... 275
*Conventions and Introduction* ..... *275*
Chapter 2 ..... 277
*File Descriptions* ..... *277*
*2.1 Processed Files* ..... *277*
*2.2 Processing Control Files* ..... *290*

*2.3 On-board Glider Information* ..... ***296***
*2.4 Command and Control Files* ..... ***299***

**APPENDIX C** *Extended PicoDOS® Reference Manual, v66.07* ..... ***303***

**APPENDIX D** *Dive Data Visualization Software* ..... ***315***

Introduction ..... **315**
Acronyms, Abbreviations, and Definitions ..... **315**
Required Tools ..... **316**
*Software* ..... ***316***
*Hardware* ..... ***316***
Installation and Configuration ..... **316**
*MATLAB 2010a* ..... ***316***
*iRobot Dive Data Analysis Software (Visualization Software)* ..... ***316***
Using the Dive Plot Analysis Software ..... **316**
*Run the MATLAB Application* ..... ***316***
*Set Current Folder* ..... ***317***
*Loading the Dive Plot GUI* ..... ***318***
*Analyzing Dive Data* ..... ***321***

**APPENDIX E** *Autonomous Self Test* ..... ***347***

**APPENDIX F** *Hardware and Configuration Menus* ..... ***367***

Parameters and Configuration Menu ..... **370**
Basic Mission and Seaglider Parameters ..... **371**
Dive Parameters ..... **372**
Flight Parameters ..... **373**
Surface Parameters ..... **374**
RAFOS Parameters ..... **375**
Password ..... **375**
Telnum ..... **376**
Altnum ..... **377**
Pitch Parameters ..... **378**
Roll Parameters ..... **379**

VBD Parameters ..... **380**
Hardware Configuration Parameters ..... **381**
Pressure (external) Parameters ..... **382**
Pressure (internal) Parameters ..... **382**
Compass Parameters ..... **383**
Altimetry Parameters ..... **384**
Sea-Bird CT Calibration Coefficients ..... **385**
Power Parameters ..... **386**
Edit All Parameters ..... **386**
Validate Parameters ..... **390**
Show Parameter Details ..... **390**
Show Changed Parameters ..... **391**
Clear Changed Parameters ..... **391**
Save Parameters By Name To A File ..... **392**
Dump Parameters To Screen ..... **392**
Load Parameters From A File ..... **393**
Reset To Defaults ..... **393**
Hardware Menu (for Tests and Monitoring) ..... **394**
Pitch Control Menu ..... **394**
Current Motor Position ..... **395**
Move to Position (AD counts and cm) ..... **395**
*Change the A/D counts* ..... ***395***
*Change the cm (centimeters) of movement* ..... ***396***
Edit Pitch Parameters ..... **396**
Run Pitch Duty Cycles ..... **397**
Run Pitch and Roll Duty Cycles ..... **398**
Pitch Test ..... **399**
Roll Control Menu ..... **400**
VBD Menu ..... **401**
Characterize Valve ..... **402**
Valve Noise Test ..... **403**
Open Valve ..... **404**
Close Valve ..... **404**
Cycle Valve ..... **404**
Pump and Bleed Cycles ..... **405**
Pump and Hold at Pressure ..... **407**
Special Test #1 ..... **408**
Supervisor Menu ..... **409**

Pressure Sensor Menu ..... 410
Compass/Attitude Menu ..... 410
GPS Menu ..... 412
Modem Menu ..... 412
Internal Pressure Menu ..... 413
Altimeter/XPDR Menu ..... 413
Sensors Menu ..... 414
Loggers Menu ..... 415
Batteries and Fuel Gauges Menu ..... 416
Low-Level Hardware Menu ..... 416
Miscellaneous Hardware Functions Menu ..... 418
Developer Tests Menu ..... 419
Test Operation Modes amd Files ..... 419
PicoDOS Commands ..... 420
Pre Launch Menu ..... 421
Set Scenario Mode ..... 421
Perform Interactive Self Test ..... 422
Perform Autonomous Self Test ..... 422
Upload Self Test Results ..... 422
Reset Dive/Run Number ..... 423
Test Launch ..... 423
Sea Launch ..... 423
Serial Port Data Transmission ..... 424

**APPENDIX G** *Ballasting* ..... ***425***

Overview ..... 425
Re-ballasting in the field ..... 426

**APPENDIX H** *Pilot and Field Team Checklist* ..... ***431***

**APPENDIX I** *Sample Field Kit Checklist* ..... ***439***

Field Kit Checklist ..... 439

**APPENDIX J** *Warranty and Disclaimers* ..... ***441***

Seaglider™ Warranty and Disclaimers ..... **441**
*1. Warranty, Exclusive Remedies and Warranty Disclaimers.*... ***441***
*2. Limitations of Liability and Disclaimer of Damages.* ..... ***442***

# *List of Figures*

**FIGURE 1-1.** 1KA Seaglider system overview ..... 27
**FIGURE 2-1.** 1KA Seaglider in Open Shipping Case ..... 30
**FIGURE 3-1.** Launch and Recovery Cradle Components ..... 38
**FIGURE 3-2.** Cradle bolts, handle and rail assembly ..... 39
**FIGURE 3-3.** Assemble Cradle ..... 40
**FIGURE 3-4.** Position of Seaglider in the Cradle ..... 41
**FIGURE 3-5.** Bulkhead Connector Locations on Aft Endcap ..... 43
**FIGURE 3-6.** Sensor Port Locations on Aft Endcap ..... 44
**FIGURE 3-7.** Fully Assembled Seaglider ..... 45
**FIGURE 3-8.** Seaglider with Antenna Mast Removed from the Aft Fairing ..... 46
**FIGURE 3-9.** Antenna in stowed position ..... 47
**FIGURE 3-10.** Connecting the cable to the antenna mast ..... 48
**FIGURE 3-11.** Connecting the cable to the laptop ..... 49
**FIGURE 3-12.** Seaglider Magnetic Wand ..... 50
**FIGURE 3-13.** Wanding Seaglider On ..... 51
**FIGURE 3-14.** Seaglider Set Date, Time, and Power Source ..... 52
**FIGURE 3-15.** Verify Primary Telephone Number ..... 53
**FIGURE 3-16.** Verify Alternate Telephone Number ..... 54
**FIGURE 3-17.** Pre-Launch ..... 55
**FIGURE 4-1.** iRobot 1KA Seaglider and Its Primary Parts ..... 60
**FIGURE 4-2.** Mass Shifter Causing Pitch Change ..... 62
**FIGURE 4-3.** Mass Shifter Causing Roll Change ..... 63
**FIGURE 4-4.** External Bladder Inflation and Deflation ..... 68
**FIGURE 4-5.** Canonical Dive Schematic ..... 74
**FIGURE 6-1.** Seaglider with Antenna Mast Removed from the Aft Fairing ..... 175
**FIGURE 6-2.** Travel Mode Menu -1 ..... 176
**FIGURE 6-3.** Travel Mode Menu – 2 ..... 177
**FIGURE 6-4.** Pre-Launch ..... 180
**FIGURE 6-5.** Sea Launch ..... 181
**FIGURE 6-6.** GPS Fix ..... 182
**FIGURE 6-7.** Final OK to Launch ..... 183

**FIGURE 7-1.** Communications Cable Attached to Seaglider ..... 189
**FIGURE 7-2.** Seaglider Positioned for Self Test ..... 190
**FIGURE 7-3.** Connecting the Communication Cable to the Laptop ..... 191
**FIGURE 7-4.** Seaglider Start-Up on the Laptop] ..... 192
**FIGURE 7-5.** Loggers Menu ..... 193
**FIGURE 7-6.** Turning off the GPCTD ..... 194
**FIGURE 7-7.** Altimeter Check ..... 196
**FIGURE 7-8.** Interactive Self-Test ..... 197
**FIGURE 7-9.** Removing Old Data/Log Files ..... 198
**FIGURE 7-10.** Seaglider ID Verification and Communications Set Up ..... 198
**FIGURE 7-11.** Checking GPS and Iridium ..... 200
**FIGURE 7-12.** Checking Bathymetry Data ..... 201
**FIGURE 7-13.** Compass Calibration ..... 201
**FIGURE 7-14.** Critical Operational Parameters ..... 202
**FIGURE 7-15.** Reporting Hardware Configuration and Software Versions ..... 202
**FIGURE 7-16.** Reporting Directory Contents] ..... 203
**FIGURE 7-17.** Checking Pressure Sensor ..... 203
**FIGURE 7-18.** Checking GPCTD ..... 204
**FIGURE 7-19.** Checking Pitch Motor] ..... 204
**FIGURE 7-20.** Checking Roll Motor ..... 206
**FIGURE 7-21.** Checking VBD Motor and Value ..... 207
**FIGURE 7-22.** Checking sensors and data file creation ..... 209
**FIGURE 7-23.** Internal Pressure and Humidity Sensors ..... 210
**FIGURE 7-24.** Checking Altimeter ..... 210
**FIGURE 7-25.** Reporting Targets and Science Specifications ..... 211
**FIGURE 7-26.** Battery Status/Fuel Gauges ..... 211
**FIGURE 7-27.** Checking Capture Vector ..... 212
**FIGURE 7-28.** Checking Flash ..... 212
**FIGURE 7-29.** Iridium Communications and Transmission Check ..... 213
**FIGURE 7-30.** Self Test Pass/Fail Status ..... 213
**FIGURE 7-31.** Loggers Menu ..... 215
**FIGURE 7-32.** Set MinCondFreq ..... 216
**FIGURE 7-33.** Example Excerpt from GPCTD Cal Sheet ..... 217
**FIGURE 7-34.** Position of Seaglider During a Simulated Dive ..... 222
**FIGURE 7-35.** Seaglider Start-Up on the Laptop ..... 223

**FIGURE 7-36.** Main Menu and Launch Menu ..... 224
**FIGURE 7-37.** Pitch, Roll, and Pump Maneuvers ..... 225
**FIGURE 7-38.** Reset Dive/Run Number ..... 225
**FIGURE 7-39.** File Upload to Seaglider ..... 226
**FIGURE 8-1.** Main Plot ..... 231
**FIGURE 8-2.** Dive 1 Pitch Plot ..... 232
**FIGURE 8-3.** $SM_CC Values ..... 234
**FIGURE 8-4.** Roll Center Values ..... 235
**FIGURE 11-1.** Post Deployment Seaglider Cleaning ..... 261
**FIGURE F-1.** Main Menu ..... 367
**FIGURE F-2.** Parameters and Configuration ..... 370
**FIGURE F-3.** Basic Mission and Seaglider Parameters Menu ..... 371
**FIGURE F-4.** Checking/Changing Seaglider's Mission Number ..... 372
**FIGURE F-5.** Dive Parameters Menu ..... 372
**FIGURE F-6.** Flight Parameters Menu ..... 373
**FIGURE F-7.** Surface Parameters Menu ..... 374
**FIGURE F-8.** RAFOS Parameters Menu ..... 375
**FIGURE F-9.** Seaglider Password Check ..... 376
**FIGURE F-10.** Verifying Basestation Telephone Number in Seaglider ..... 376
**FIGURE F-11.** Verifying Basestation Alternate Telephone Number in Seaglider ..... 377
**FIGURE F-12.** Pitch Parameters Menu ..... 378
**FIGURE F-13.** Roll Parameters ..... 379
**FIGURE F-14.** VBD Parameters Menu ..... 380
**FIGURE F-15.** Hardware Configuration Parameters Menu ..... 381
**FIGURE F-16.** External Pressure Parameters Menu ..... 382
**FIGURE F-17.** Internal Pressure Parameters Menu ..... 383
**FIGURE F-18.** Compass Parameters Menu ..... 383
**FIGURE F-19.** Altimetry Parameters Menu ..... 384
**FIGURE F-20.** Sea-Bird CT Calibration Coefficients Menu ..... 385
**FIGURE F-21.** Power Parameters Menu ..... 386
**FIGURE F-22.** Edit All Parameters Menu ..... 387
**FIGURE F-23.** Save Parameters By Name To A File ..... 392
**FIGURE F-24.** Hardware Menu ..... 394
**FIGURE F-25.** Pitch Control Menu ..... 394
**FIGURE F-26.** Pitch Motor Movement By Changing A/D Counts ..... 396

**FIGURE F-27.** Pitch Duty Cycling ..... 398
**FIGURE F-28.** Roll Control Menu ..... 400
**FIGURE F-29.** VBD Control Menu ..... 401
**FIGURE F-30.** VBD Valve Characterization ..... 402
**FIGURE F-31.** Valve Noise Test ..... 403
**FIGURE F-32.** VBD Valve Cycling ..... 405
**FIGURE F-33.** VBD Pump and Bleed Pressure Test ..... 406
**FIGURE F-34.** VBD Pump and Hold at Pressure Test ..... 407
**FIGURE F-35.** Special Test #1 ..... 409
**FIGURE F-36.** Supervisor Menu ..... 409
**FIGURE F-37.** Pressure Sensor Menu ..... 410
**FIGURE F-38.** Compass Menu ..... 411
**FIGURE F-39.** GPS Menu ..... 412
**FIGURE F-40.** Iridium Phone Menu ..... 413
**FIGURE F-41.** Internal Pressure Menu ..... 413
**FIGURE F-42.** Altimeter/XPDR Menu ..... 414
**FIGURE F-43.** Sensors Menu ..... 414
**FIGURE F-44.** Example Menu ..... 415
**FIGURE F-45.** Loggers Menu ..... 415
**FIGURE F-46.** Batteries and Fuel Gauges Menu ..... 416
**FIGURE F-47.** Low-level Hardware Menu ..... 417
**FIGURE F-48.** Miscellaneous Menu ..... 418
**FIGURE F-49.** Developer Tests Menu ..... 419
**FIGURE F-50.** Test Operation Modes and Files ..... 420
**FIGURE F-51.** Pre Launch Menu ..... 421
**FIGURE A-1.** Ballast Location Pin-wheel ..... 428

# *Preface*

This section provides an overview of the iRobot® 1KA Seaglider™ Unmanned Underwater Vehicle. The following topics are covered:


- "Who Should Use this Manual" on page 17
- "Typographical Conventions" on page 18
- "Safety Conventions" on page 19
- "Safety Information" on page 19
- "Registration" on page 20
- "iRobot Contact Information" on page 20


## *Who Should Use this Manual*

This manual is for qualified and authorized Seaglider pilots and field personnel only. It provides an overview of the operating principles and instructions for assembling, testing and piloting Seagliders as well as basic guidelines for launching and recovering. Launch and recovery methods vary with company and facility.

## *Typographical Conventions*

The following table explains the typographical conventions used in this manual.

| Typeface | Indicates | Example |
|---|---|---|
| **UPPERCASE BOLD** | Parameters used to control the operations of the 1KA Seaglider; includes a leading **$** sign. | (**$T_DIVE**) |
| **bold** | Items you select, (such as menu options and commands), type on screen, or choose from a list. | Type **cmdfile** and then click **Edit**. |
| monospace | User input and filenames. | /etc/vx/tunefstab |
| *italic* | Variables, titles, first occurrence of terms. | *Variable name*. |
| CAPITALS | Names of keys on the keyboard | SHIFT, CTL, or ALT |
| **Tab>Command Group>Command** | Command option paths are written in the order you access a command button from the ribbon | Edit>Insert>Insert file |
| ENTER | Press the Enter key on the keyboard when <CR> appears on the screen. | Type **1** for Parameters and Configuration, and then press ENTER. |

## *Safety Conventions*

Seaglider is designed to be as safe as possible to the user and the operating environment. Common sense and good judgment will help prevent injuries and damage to the equipment. Make sure to read the Warnings, Cautions, and Notes carefully.

**Warning:** This symbol indicates a potential hazard to personnel.

**Caution:** This symbol indicates that the user should refer to the instruction manual in order to protect the apparatus from damage.

**Note:** Note statements contain information that requires special attention from the operator.

## *Safety Information*

Be aware of the following types of safety hazards:

### Personal Safety Information

**Warning:** As shipped, Seaglider's chassis weighs approximately 120 lbs (52 kg) dry and is capable of inflicting personal injury to body parts. When removing Seaglider from the shipping container, use **extreme caution** to avoid personal injury or equipment damage.

Seaglider should be carried by two people. Each person should hold one end of the cradle. To prevent back injury, lift with your legs instead of your back.

### Vehicle Hazards

**Warning:** Before performing maintenance procedures, make sure that the vehicle is powered off. Serious injury can occur if the 1KA Seaglider's power supply is intact while maintenance is being performed.

### Battery Safety

Seaglider uses (1) 24V lithium primary battery pack and (1) 10V lithium primary battery pack.

**Warning:** Release of toxic gases/materials due to battery exposure to fire can cause death or injury.

## *Registration*

To access user information on the iRobot web site, you must register your Seaglider. To register your product and establish a user ID and a password, go to the iRobot web site at:

http://gisupport.irobot.com/app/utils/login_form

## *iRobot Contact Information*

### Customer Support

For iRobot Customer Support, call the Support Phone Line: 1-781-430-3030. The Support Team is available by phone during normal business hours (Monday through Friday, 0800-1700 EST). After normal business hours, leave a voicemail message; the Support Team responds to all voicemail messages the next business day.

Access the iRobot Customer Support web site at the following address:

http://gisupport.irobot.com/cgi-bin/irobot_gi.cfg/php/enduser/acct_login.php

### Sales

To obtain pricing information or to order spares, options, accessories, or training, send an e-mail to sales@irobot.com or call 888-776-2687 (Toll free in the USA) or +(1) 781-430-3090 (International customers).

CHAPTER 1

# 1KA Seaglider

This chapter provides an overview of the 1KA Seaglider history and components. The following topics are covered:


- "Seaglider History" on page 23
- "System Overview" on page 25
- "Seaglider Sensors" on page 27


## Seaglider History

The history of buoyancy-driven oceanographic instruments begins with Archimedes (287 BCE - 212 BCE). He is considered the father of hydrostatics, static mechanics, and integral calculus. Archimedes' Principle is the basis for all buoyancy driven vehicles. It states that the buoyant (upward) force on a submerged object is equal to the weight of the fluid that the object displaces. This fact is used in the variable mass, fixed volume (ballast) control systems of modern submarines and submersibles, and in the fixed mass, variable volume control systems of small profiling oceanographic instruments.

The use of buoyancy control in oceanographic instruments dates from the mid-1950s. By 1955, Henry Stommel of the Woods Hole Oceanographic Institution and

John Swallow in the United Kingdom had ideas for neutrally buoyant floats whose positions could be tracked acoustically. Swallow was the first to build such a device, which contained a free-running 10 kHz acoustic source and was tracked from a surface ship. By the 1970s, transponding versions running at 3-4 kHz had extended shipboard detection ranges to 50 km, and a 200 Hz version used the Sound Fixing and Ranging (SOFAR) sound channel (Stommel's original idea) to remove the requirement for ship-based tracking.

By the 1980s, Tom Rossby at URI had developed the inverse of the SOFAR float (called RAFOS, SOFAR spelled backwards) that relied on moored sound sources and an acoustic receiver on the float. By adding a compressee (an object whose compressibility is approximately the same as that of seawater), these floats could also be ballasted to follow a particular density surface, rather than a pressure surface. About the same time, John Dahlen's group at Charles Stark Draper Laboratory developed a moored profiler that used a variable buoyancy device to propel itself up and down along the mooring wire, measuring temperature, conductivity and currents.

In the 1990s, Russ Davis and his group at Scripps Institution of Oceanography added a variable-buoyancy device to a neutrally buoyant float to create profiling floats. These floats (called Autonomous Lagrangian Current Explorers, or ALACE) had the ability to inflate an external bladder, thereby changing their displaced volume, but not their mass. The resulting buoyancy force allowed the float to make profile measurements from its neutrally buoyant depth to the surface. At the surface, position and profile data were transmitted via the Service ARGOS satellite system. By the year 2000, hundreds of this type of float were deployed worldwide, both of the Scripps design and a design from Webb Research Corporation (now Teledyne Webb Research) of East Falmouth, Massachusetts.

Today's gliders share a common heritage: Henry Stommel's vision, published in 1989 in Oceanography [Stommel, 1989]. Stommel imagined a fleet of vehicles that "...migrate vertically through the ocean by changing ballast, and they can be steered horizontally by gliding on wings. During brief moments at the surface, they transmit their accumulated data and receive instructions. Their speed is about 0.5 knot." A prototype gliding vehicle was fielded as early as 1991 by Webb Research Corporation (WRC). This vehicle demonstrated the basic configuration of future gliders.

A few years later, the Office of Naval Research-sponsored Autonomous Ocean Sensing Network (AOSN) program, led by Tom Curtin, sponsored three groups to develop autonomous underwater gliders.

All groups worked with similar design goals:

- Small enough to be handled by two people
- Relatively low acquisition and operation costs
- Horizontal speeds of around 30 cm/s
- Endurance of up to a year
- GPS positioning
- Two-way data telemetry at the surface
- Basic sensor payloads, including a Conductivity, Temperature, and Density

By the year 2000, all groups had operational models that addressed these design goals:

- Slocum: developed by Webb Research Corporation
- Spray: developed by a team of scientists from Scripps Institution of Oceanography (Russ Davis) and Woods Hole Oceanographic Institution (Breck Owens)
- Seaglider: developed at the University of Washington (Charlie Eriksen)

The development of the autonomous Seaglider has greatly extended the density of hydrographic observations at orders of magnitude lower cost than is possible with ships and moorings. The construction cost of a Seaglider is equivalent to a few days of a UNOLS open ocean ship time and its annual operational cost is less than a day of the same.

This user guide describes operating Seaglider, the vehicle developed by the University of Washington School of Oceanography and the Applied Physics Laboratory, now licensed to iRobot Corporation for manufacture.

## *System Overview*

Seaglider is an Unmanned Underwater Vehicle (UUV) designed for use in oceanographic missions lasting up to 10 months and covering up to 6000km at depths ranging from 50 to 1000m.

Seaglider travels underwater in a sawtooth pattern. The vertical velocity component of the sawtooth pattern comes from the onboard buoyancy engine changing

Seaglider's density while the horizontal velocity component comes from the lift provided by Seaglider's wings and, to a much lesser extent, Seaglider's body.

Seaglider runs on the PicoDOS operating system and can be reprogrammed and redirected from its basestation at any time, receiving the new commands when it surfaces.

Figure 1-1 shows the parts (land side and field side) of Seaglider's system and how they communicate with one another via Iridium satellite.

The field side consists of the following:

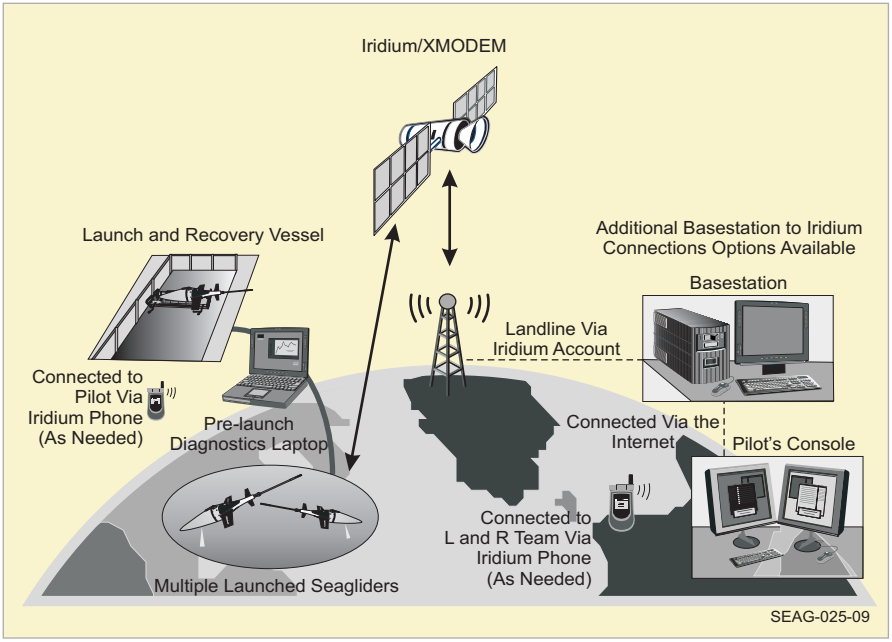
- Seaglider
- Pre-flight diagnostic laptop used to check out Seaglider before deployment
- Field team

The land side consists of the following:

- Basestation CPU running the Linux® operating system
- Modem
- Pilot team
- Basestation interface computer

Seaglider makes a phone call, and the signal is picked up by the Iridium satellite. The call is then downloaded to the civilian Iridium ground station in Arizona where it is then sent via landline (phone line connected to the modem) to the basestation.

**FIGURE 1-1. 1KA Seaglider system overview**




## *Seaglider Sensors*

Seaglider is equipped with third-party sensors that measure conductivity, temperature, pressure, dissolved oxygen, fluorescence, Chromophoric Dissolved Oxygen Matter (CDOM) and optical backscatter. Using satellite based communications, the system transmits collected data to the basestation each time it surfaces. The following sections describe the sensing devices.

### Standard Seaglider Sensors

The following third-party sensors are installed as standard equipment in Seaglider:

- PAINE ® strain-gauge pressure sensor

### Optional Sensors

The following optional third-party sensors have been installed in select Seagliders:

- Sea-Bird 43F dissolved oxygen sensor (pumped)
- Sea-Bird dissolved oxygen sensor (unpumped)
- Sea-Bird CT Sail
- Aanderaa dissolved oxygen
- WET Labs ECO Pucks™
- Photosynthetically Active Radiation (PAR) sensor
- Payload Conductivity Temperature Density (GPCTD) sensor

The pilot controls the frequency of science data collection by specifying sensor sampling intervals.

CHAPTER 2

# *Checking Out Crate Contents*

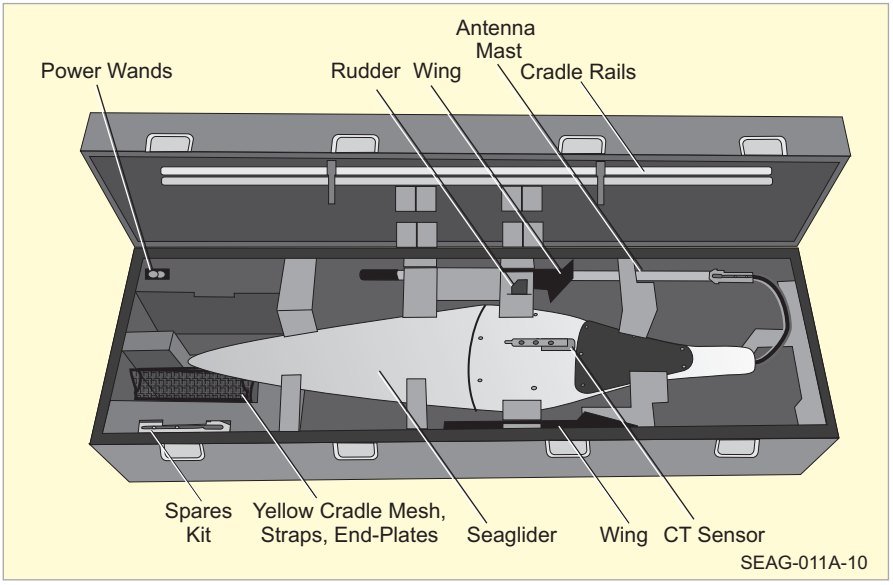
This chapter introduces the 1KA Seaglider standard components and optional equipment. The following topics are covered:


- "Inspecting Seaglider's Shipping Container" on page 30
- "Inspecting the Basestation Container" on page 31

## *Inspecting Seaglider's Shipping Container*

Seaglider is shipped in a reusable container that serves as a field transport and storage case. Your configuration may differ, depending on what you ordered.

**FIGURE 2-1. 1KA Seaglider in Open Shipping Case**




Locate the following in the shipping case:

- 1KA Seaglider body
- Sea-Bird Electronics conductivity and temperature (CT) sensor (installed on Seaglider body)
- Port and starboard wings (labeled)
- Rudder
- Antenna mast attached to rudder shoe
- Seaglider notebook containing trim sheets, sensor documentation, Seaglider serial number, and transponder frequency
- Two magnetic power on/off wands
- Plastic spare parts box containing:
  - #2 Phillips head screw driver

- Spare antenna connector ‘o’-rings
- Two spare lead trim weights (5” long x ¾" wide x 1/8" thick)
- Small vice grips
- All screws necessary for assembly and spares

Optional Components:

- Laptop computer for Seaglider communications
- 50 ft. non-powered communications cable
- Powered communications cable
- Launch and recovery cradle components (Figure 3-1)
  - Four aluminum rails
  - Yellow mesh
  - Two (2) end plates with carrying handles
  - Four (4) nylon straps (2 male, 2 female)
  - Assembly screws (8), washers (8), lock washers (8), wing nuts (4), eye screws(4) in a plastic bag.

If damage to your Seaglider occurred during transit, or if your shipment is incomplete, please contact iRobot Maritime customer support at 781-430-3030, ext. 2.

## *Inspecting the Basestation Container*

Locate the following in the basestation container, if you purchased the basestation from iRobot:

- CPU with Linux installed (prefer Fedora core 10 or 11)
- Monitor
- Power cord
- Keyboard
- Mouse
- Two (2) port serial controller
- Two (2) modems with cables and power supplies

If damage to your basestation occurred during transit, or if your shipment is incomplete, please contact iRobot Maritime customer support at 781-430-3030, ext. 2.

**Note:** If you did not purchase the basestation from iRobot, you must provide a system that contains the basestation items listed above.

CHAPTER 3

# *Setting Up the System*

This chapter discusses the set-up of the basestation, the Pre-flight Diagnostic Computer (field laptop), the interface computer, the cradle to hold Seaglider, and Seaglider.

The following topics are covered:


- "Setting Up the Basestation" on page 34
- "Setting Up Seaglider's Interface Computer" on page 37
- "Setting Up the Pre-Flight Diagnostic Laptop" on page 37
- "Assembling Seaglider's Cradle" on page 38
- "Assembling Seaglider" on page 41
- "Checking Out the Seaglider System" on page 47

## *Setting Up the Basestation*

Seaglider's basestation is the shore-side computer end of the Seaglider system. It is the computer that gliders call through the Iridium phone system. It has three main functions:

- Supports a modem (or modems) and dial-up users (Seagliders)
- Handles one side of the modem-to-basestation transfer protocol
- Processes Seaglider's data, producing scientific and engineering data, and performs simple error detection and notification.

The pilot does not have to sit at the basestation to control Seaglider, although they can. Often the pilot accesses the basestation over the internet from another computer, Seaglider's Interface Computer (see "Setting Up Seaglider's Interface Computer" on page 37). This setup allows the pilot to control Seaglider from anywhere there is internet access while Seaglider calls into the stationary basestation over the dedicated phone lines or internet via the RUDICS interface.

### Basestation Configuration

Seaglider's basestation runs on a Linux® operating system (OS ), preferably Fedora core 13. The basestation software package consists of a collection of python scripts, a patched version of the XMODEM send and receive programs and configuration support for the modem program MGETTY.

The basestation is configured to auto-answer dial-up calls. Seagliders log in as normal dial-up users, and then send and receive files to/from their home directory on the basestation.

Seaglider pilots need to have write access in those Seaglider home directories to modify command and control files. At Seaglider login and logout, scripts are run to control and record aspects of the basestation transactions.

### Basestation Log In

To log into the basestation remotely, you need a user name, a password and the IP address of the basestation. When the basestation arrives from iRobot, the user name is set to *pilo*t and the initial password is *1qazxsw2*. It is recommended that you change this password immediately using the 'passwd' command.

### Internet Configuration

To connect the basestation to the internet you need a static IP address. Set up the basestation on the local network, assign it an IP address, and connect it to a dedicated phone line (no extensions) through the modem. If you need help, contact your IT department. If RUDICS is being used, the RUDICS port (configured by the user) will need to be opened through the firewall on the local machine as well as any firewalls between the basestation and the wide open internet.

### Setting Up the Basestation Directories and Files

After logging in to the basestation and connecting it to the internet, there are several things that you must check and set up.

#### *Verify that Seaglider is Commissioned on the Basestation*

To verify whether the Seaglider you are working with has been commissioned on the basestation:

1. Type **cd /home**, and then press ENTER.
2. Type **ls**, and then press ENTER.

   If the Seaglider has been commissioned on the basestation, a directory with that Seaglider's name (for example, sg505) displays. All Seaglider names start with sg. The three digit number in the name is Seaglider's serial number. Seaglider's name is in the manual shipped with Seaglider. In addition, the serial number is written in black on the rear of the aft fairing.
3. If the Seaglider is not listed, create a directory for the Seaglider (see "Commissioning a Seaglider on the Basestation" on page 35). Otherwise, see"Checking the Contents of Seaglider's Directory" on page 36 for more information.

#### *Commissioning a Seaglider on the Basestation*

To create a directory on the basestation:

1. Type **su**, and then press ENTER.

   **Note:** You must be logged in as super user (root) to create a directory.
2. When prompted, type the *password* for super user (i.e. super user (root)).

   The initial password is 1qazxsw2 (the number one, followed by qazxsw, and then the number 2).
3. Type **python /usr/local/basestation/Commission.py** *XXX* and then press ENTER, where *XXX* is the serial number of the Seaglider being commissioned.

4. Type **ls**, and then press ENTER.

   The new directory displays. If it does not, repeat step 3.
5. When finished, type **exit**, and then press ENTER.

   **Note:** DO NOT stay logged in as root.

### *Checking the Contents of Seaglider's Directory*

If the Seaglider directory is already present on the basestation (commissioning not needed), check the contents of the Seaglider directory on the basestation.

**Note:** Check the contents of the Seaglider directory at the start of every new set of testing and every mission.

To check the contents of the Seaglider's directory:

1. Type **cd** *~sgXXX*, and then press ENTER.

   where XXX is the Seaglider serial number.
2. Type **ls***,* and then press ENTER. to check the directory's contents.
3. If there are any old data files (.asc, .cap, .dat, .eng, .log, .prm), move them to a new subdirectory with a descriptive name. See Chapter 3, "Moving Data Files" on page 36 for information.

   The cmdfile, science, targets, .pagers, .urls and sg_calib_constants.m files should remain in the sgXXX home directory. See Chapter 9, "Files for Operations" on page 237 for a description of the contents of the files.

### *Moving Data Files*

The program movedata.sh, that creates a subdirectory named by the user and moves older files from the top directory to that subdirectory, was installed at the factory.

To move data files:

1. Type **/usr/local/basestation/movedata.sh** (sub-directory name of the your choice).
2. Press ENTER.

   For example, type **movedata.sh BuzzardsBay_12Jan10**, and then press ENTER.

### *Setting Up a Visualization Site*

If you have a visualization site for the Seaglider data, set up the communication and data transfer between the basestation and the visualization site. Edit the .url file on the basestation to send data to the visualization site. This set up is site/user specific. Your IT department can set it up.

## *Setting Up Seaglider's Interface Computer*

The basestation interface computers are ones that pilots can use to connect to the basestation over the internet. Compared to the basestation, there is much more flexibility in how they are set up. Any OS is acceptable including Linux, Windows or Mac OS X. Any software that provides Secure Shell (SSH) and Secure File Transfer Protocol (SFTP) is acceptable as well.

**Note:** The program SSH (Secure Shell) is a secure replacement for telnet and the Berkeley r-utilities (rlogin, rsh, rcp and rdist). It provides an encrypted channel for logging into another computer over a network, executing commands on a remote computer and moving files from one computer to another. SSH provides strong host-to-host and user authentication as well as secure encrypted communications over an insecure Internet.

iRobot usually uses Windows for the OS, PuTTY (open source) for the SSH client, and Windows Secure CoPy (WinSCP) as the SFTP client. However, there are dozens of programs that you can use on all three operating systems, so use whatever program you want. After loading the appropriate software on the interface computer, access the basestation using the SSH and SFTP clients. Your IT department provides the host name and password.

## *Setting Up the Pre-Flight Diagnostic Laptop*

Seaglider's pre-flight diagnostic laptop (hereafter referred to as the field laptop) is the computer that connects directly to Seaglider's serial communications port. Like the basestation interface computer, there are many possible configurations. The Seaglider field laptop must have a serial port (or USB serial port adapter) and a serial terminal emulation program. iRobot uses the following:

- Operating system (OS): Windows

- Serial terminal emulation program: Tera Term Pro 2.3 (free software terminal emulator)

Configure Tera Term Pro 2.3 for 9600, 8, N, 1, no hardware handshake, echo off and <CR> only. VT100 is the terminal that is emulated.

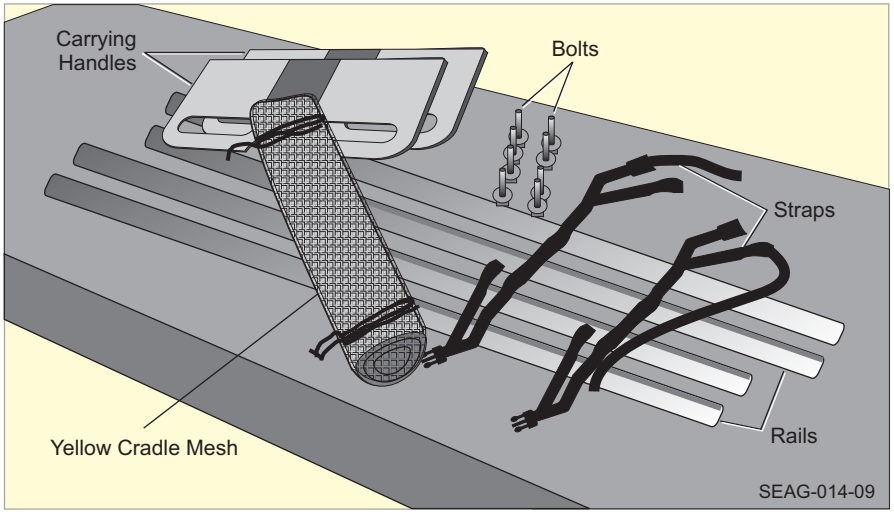
**Caution:** Do not use Windows Hyper Terminal, a serial terminal emulation program, under any circumstance. It does not function well with Seaglider.

## *Assembling Seaglider's Cradle*

To assemble the cradle:

1. Make sure that you have all the parts shown in Figure 3-1 and listed on page 31.

**FIGURE 3-1. Launch and Recovery Cradle Components**

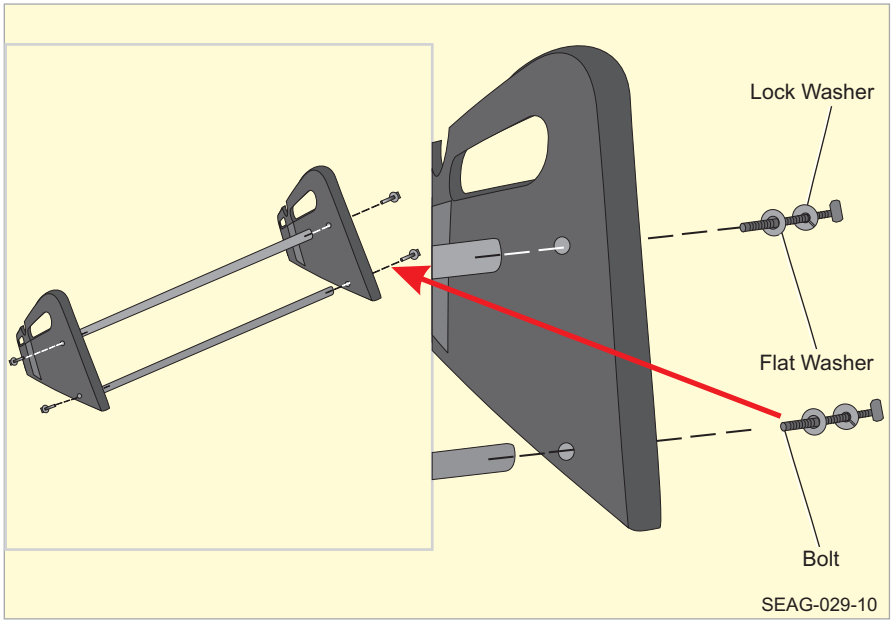



2. Insert two of the aluminum rails through the side hems of the yellow mesh (one rail per side).

   **Note:** The four aluminum rails are interchangeable.

3. Put a female strap on each end of one of the upper rails and a male strap on each end of the other upper rail.
4. Attach the top rails to the carrying handles by sliding a lock washer and then a flat washer (in that order) onto the large bolt, and then inserting the bolt from the exterior side of the bolt hole in the handle into threaded hole in the rail (see Figure 3-2).

**FIGURE 3-2. Cradle bolts, handle and rail assembly**

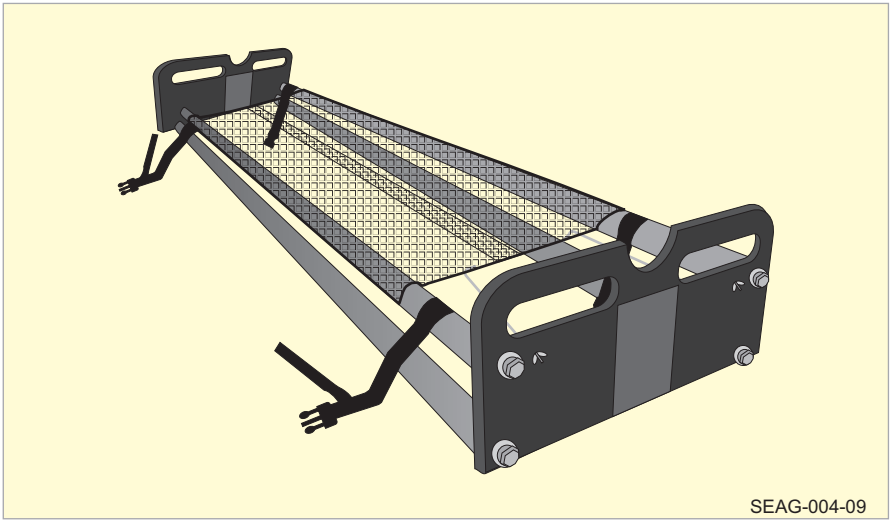



5. Give the bolts a couple of turns, but do not tighten yet.
6. Attach the lower rails to the handles using the same bolt/lock washer/flat washer combination as for the upper rails.
7. Tighten all four rails to the handles using a 3/4" wrench (not provided).
8. From the inside of the carry handles, insert the four eye bolts into the small holes on the carrying handles and secure with the wing nuts.
9. Thread the free ends of the strings attached to the mesh through their respective eye bolt and knot in place.

   The strings should be tight enough to keep the mesh from sliding more than a couple of inches in either direction, but the mesh should not be taut.

Figure 3-3 shows the assembled cradle.

**FIGURE 3-3. Assemble Cradle**



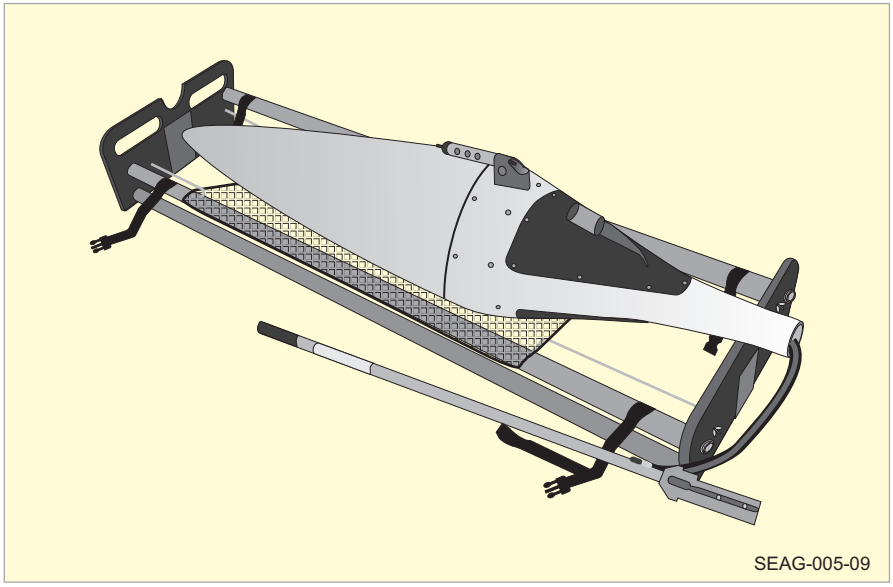

**10.** Put the assembled cradle on the floor next to the shipping case.

## *Assembling Seaglider*

To assemble Seaglider:

1. Remove Seaglider from the shipping case (a two person task) and place in the launch and recovery cradle with the tail of the aft fairing resting on one handle (see Figure 3-4). Your configuration may differ, depending on what you ordered.

**FIGURE 3-4.** Position of Seaglider in the Cradle




2. Remove the wings from case.

   Each wing is marked on the base of the root with the position where it should be secured, port or starboard side of the Seaglider, with the iRobot logo facing up and forward (see Figure 3-7 on page 45).

3. Install the port wing by inserting (8) 8-32 x 1/2" screws through the port wing and into the portside aft fairing holes.

   Start with the top screws, keeping them loose until all screws are started, and then tighten all of the screws until hand tight.

   **Note:** DO NOT over tighten the screws. Rotate Seaglider slightly to attach the lower screws.

4. Install the starboard wing by inserting (8) 8-32 x 1/2" screws through the starboard wing and into the starboard side aft fairing holes.

   Start with the top screws, keeping them loose until all screws are started, and then tighten all of the screws until hand tight.

   **Note:** DO NOT over tighten the screws. Rotate Seaglider slightly to attach the lower screws.

5. Check the wing orientation.

   The wings are correctly installed if the aft edges form a straight line across the fairing that is perpendicular to the axis of the Seaglider (see Figure 3-7 on page 45).

6. In preparation for installing the antenna and rudder, slide Seaglider back in the cradle so that the slot in the rear of the aft fairing is outside of the cradle (Figure 3-4 on page 41).

   **Note:** Seaglider is shipped from iRobot with the antenna and serial cables attached. The serial connections should be hand tight. The antenna connection should be wrench tight (finger tight, then a ¼ turn using the wrench). To access the connectors to check their tightness follow step 7 below.

   **Note:** The serial connections should be hand tight. DO NOT over tighten.

   **Caution:** Over-tightening can cause water leakage, connectivity and communications issues.

   **Caution:** Be sure to tighten serial connections prior to deployment. If they are not tightened, a loss of equipment could result.

   If you ever need to disconnect the antenna cable from the aft endcap, make sure when reattaching the cable to the endcap, that the O-ring is present inside the cable end of the antenna. If the O-ring is not present, find the spare O-ring in the plastic spares kit, lubricate it with silicon grease, such as DC4, and install. The O-ring is critical for a watertight seal. Failure to install the O-ring properly can result in loss of Seaglider.

7. Remove the screws from the black plate on the top of the aft fairing and take off the plate to expose the antenna and communication cables and the bulkhead connectors on the aft endcap and check the tightness of the connectors.

   See Figure 3-5 on page 43 and Figure 3-6 on page 44 respectively, for the location of the bulkhead connectors on the aft endcap.

8. Put the plate and the screws in a secure place near Seaglider.

**FIGURE 3-5. Bulkhead Connector Locations on Aft Endcap**

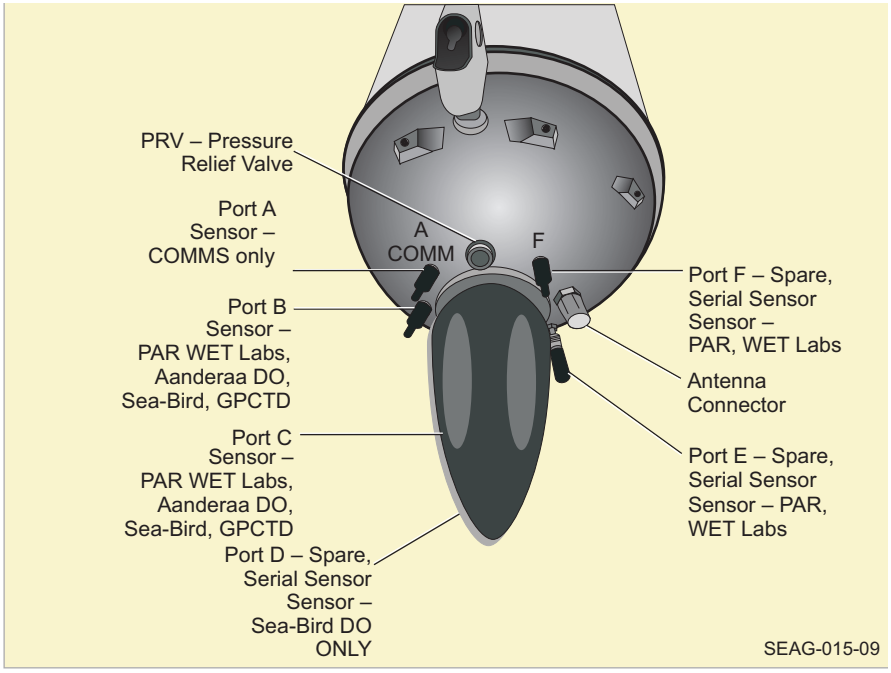
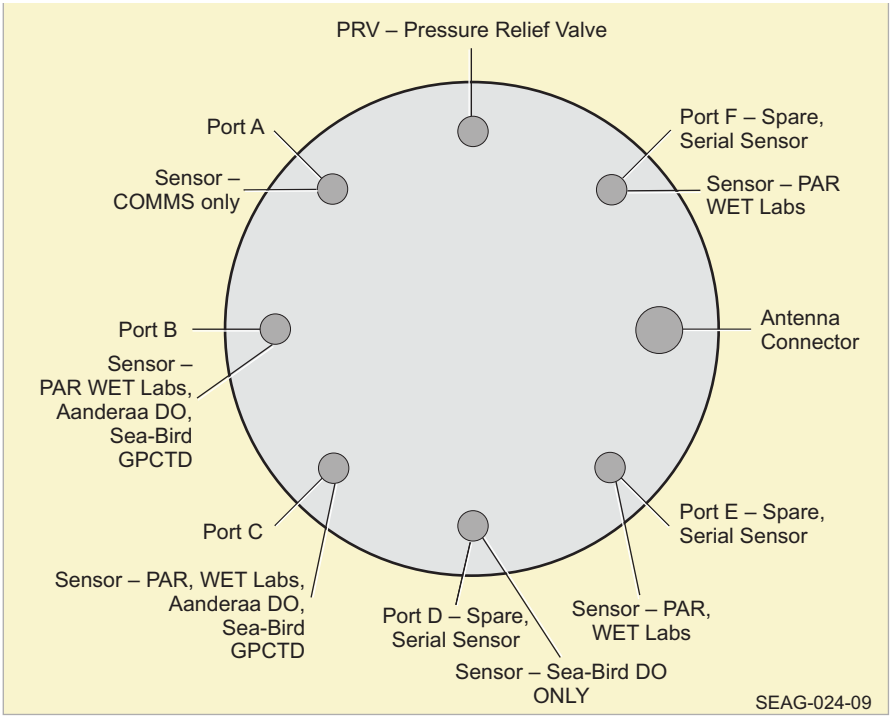

PRV – Pressure Relief Valve
Port A Sensor – COMMS only
Port B Sensor – PAR WET Labs, Aanderaa DO, Sea-Bird, GPCTD
Port C Sensor – PAR WET Labs, Aanderaa DO, Sea-Bird, GPCTD
Port D – Spare, Serial Sensor Sensor – Sea-Bird DO ONLY
A COMM
F
Port F – Spare, Serial Sensor Sensor – PAR, WET Labs
Antenna Connector
Port E – Spare, Serial Sensor Sensor – PAR, WET Labs
SEAG-015-09

**FIGURE 3-6. Sensor Port Locations on Aft Endcap**

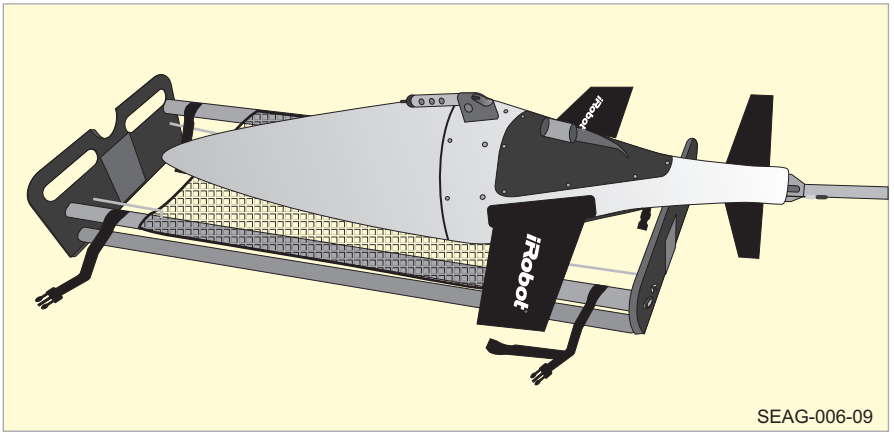

PRV – Pressure Relief Valve
Port A
Sensor – COMMS only
Port F – Spare, Serial Sensor
Sensor – PAR WET Labs
Port B
Sensor – PAR WET Labs, Aanderaa DO, Sea-Bird GPCTD
Antenna Connector
Port C
Sensor – PAR, WET Labs, Aanderaa DO, Sea-Bird GPCTD
Port D – Spare, Serial Sensor
Sensor – Sea-Bird DO ONLY
Port E – Spare, Serial Sensor
Sensor – PAR, WET Labs
SEAG-024-09

9. Remove the rudder from the packing case and (2) ¼-20 screws from the spares kit and put in a secure place near Seaglider.
10. Slide the antenna mast/rudder boot into the back end of the aft fairing, orienting the round side of the boot away from the countersunk holes in the fairing. As the antenna mast is lowered into the fairing, reach through the plate opening and guide the antenna and serial communication cables around the bladder and sensor cables.
11. When the antenna mast is in place, insert the rudder (flat edge to the rear of Seaglider, notches toward the wings) through the slit just forward of the back end of the aft fairing, making sure that the antenna and serial communication cables are not in the way.

12. Do the following:
    a. Insert (2) 1/4 x 20, 2" screws through the countersunk holes in the fairing and into the threads in the rudder base.
    b. Double check the position of the antenna cables and rearrange if they are in the way of the rudder or screws.
    c. Tighten the screws to hand tight.
13. Arrange the antenna and sensor cables in the area normally covered by the black plate so that they do not impede the operation of the external bladder.
14. Replace the black plate, starting all the screws before tightening them down to hand tight.

    Figure 3-7 shows the fully-assembled Seaglider. Your configuration may differ, depending on what you ordered.

**FIGURE 3-7. Fully Assembled Seaglider**



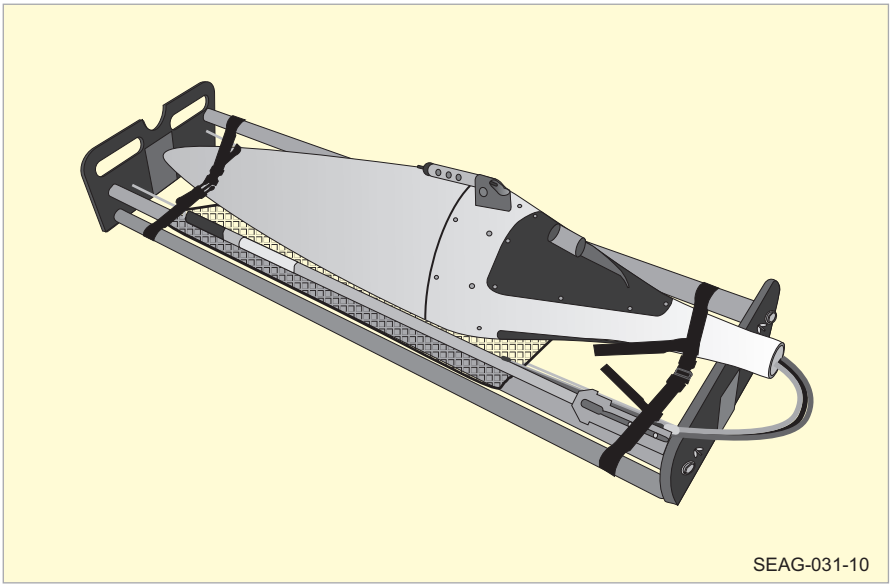

15. Slide Seaglider forward in the cradle until the nose is touching the rubber at the front of the cradle.

    **Note:** The cradle is the same on both ends. The front and back of the cradle are determined by the position of Seaglider in the cradle.

16. Connect the nylon straps and tighten to secure Seaglider in the cradle.

17. To store or move Seaglider in its cradle the antenna mast should be detached from the aft fairing following the steps below.
    a. Remove the rudder screws.
    b. Slide the antenna mast and rudder boot out of the aft fairing (can best be facilitated if the rudder section is outside of the cradle as in Figure 3-7 on page 45).
    c. Fold the antenna mast back onto Seaglider's wing and secure it with protective foam and a bungee cord.
    d. Leave the rudder in its aft fairing slot and reinstall the rudder screws to hold it in place.
    e. Seaglider should look like the one in Figure 3-8. Your configuration may differ, depending on what you ordered.

**FIGURE 3-8. Seaglider with Antenna Mast Removed from the Aft Fairing**




## *Stowing/Moving Seaglider*

See "Transporting Seaglider to the Field" on page 174.

# *Checking Out the Seaglider System*

Now that all of the pieces of the Seaglider system have been set up (the Seaglider, basestation, basestation interface computer and the field laptop), it is time to start testing them together. Hardware checkouts are done to make sure Seaglider is functional after any work is done on Seaglider, after shipping and before going into the field. The end to end checkout tests Seaglider and the communications between Seaglider and the basestation.

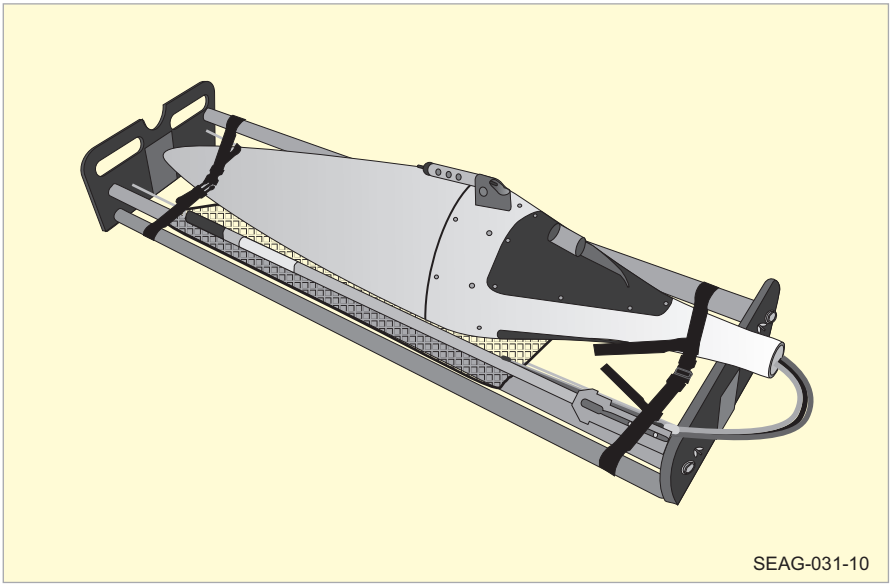
## Checking Seaglider Communication with the Field Laptop

**Note:** This test can be done in the lab.

To check Seaglider communication with the field laptop:

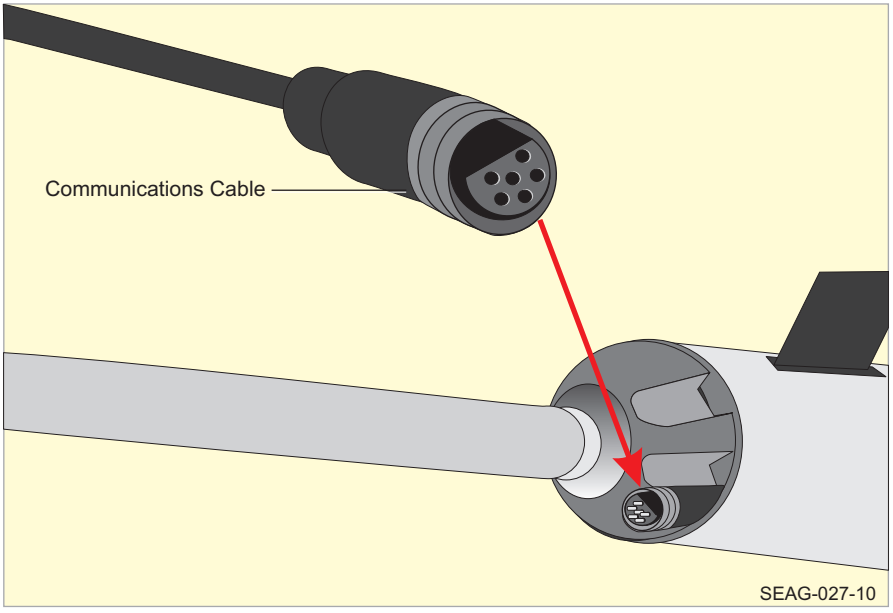
1. For this test, the antenna mast may be bolted into position as it would be for deployment (Figure 3-1 on page 38) or folded back onto Seaglider's wing in the stowed position (Figure 3-9). Your configuration may differ, depending on what you ordered.

**FIGURE 3-9. Antenna in stowed position**



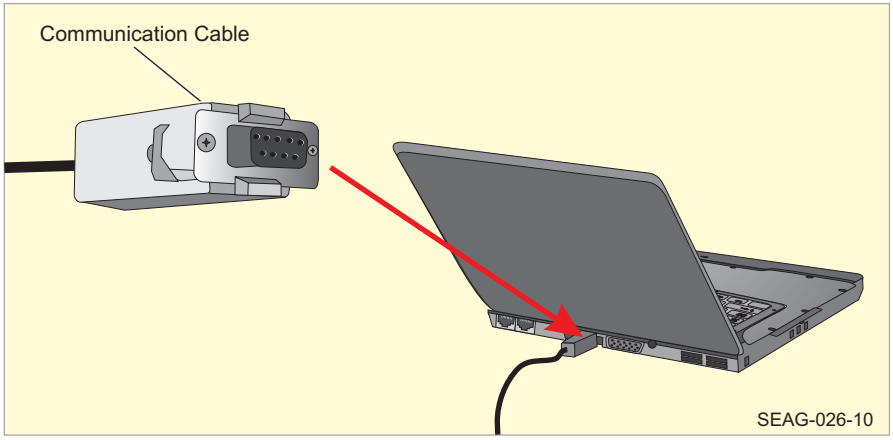
2. Do the following to connect Seaglider to the laptop via the supplied non-powered serial communication cable:
   - Connect the 6-pin IE55 end of the cable to the communication port located at the base of Seaglider's antenna mast (Figure 3-10).

**FIGURE 3-10. Connecting the cable to the antenna mast**




   - Connect the DB9 end of the cable to the laptop serial port (Figure 3-11).

**FIGURE 3-11. Connecting the cable to the laptop**




3. Turn on the laptop and start the terminal emulation program.

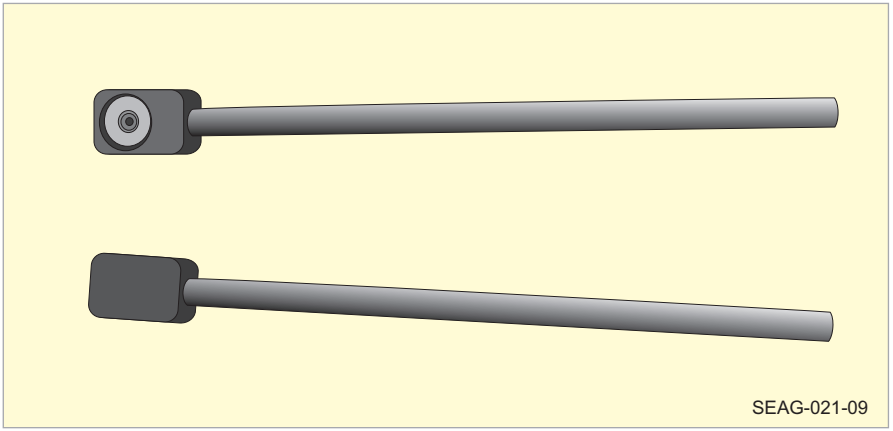
   The port settings for the emulation program are 9600, 8, N, 1, no hardware handshake, local echo off and ENTER to <CR>.

4. Turn on the screen capture on the laptop, if it is not already on.

   The method to do this varies with the terminal emulation program used.

5. Take one of the magnetic wands from the shipping case (Figure 3-12).

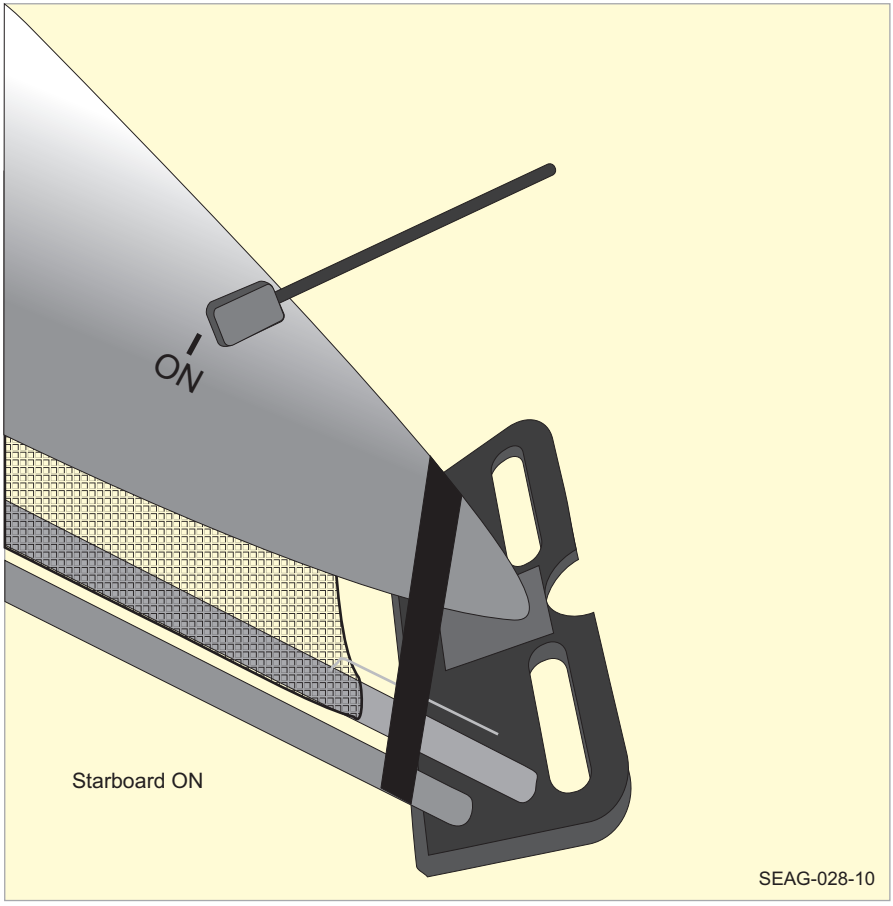
**FIGURE 3-12. Seaglider Magnetic Wand**




6. Slowly move the magnetic wand over the ON symbol on the starboard side of Seaglider, 20" aft of the nose, for at least 1 second to turn Seaglider on.

   Make sure that the wand is touching the fairing while moving it over the ON symbol (see Figure 3-13 on page 51). Several seconds after being turned on, Seaglider responds by sending lines of output to the laptop screen. If you do not see any output on the laptop screen, move the wand slowly around the mark, making a slight outward spiral, until you do.

**FIGURE 3-13. Wanding Seaglider On**




**Note:** Hold the wand on the starboard side to turn Seaglider on, and on the port side to turn Seaglider off. Use the mnemonic "Right ON!" to remember which side is On and Off.

7. Within one minute of seeing output on the screen, press ENTER.
8. In response to the date and time query, set the current date and time.

   Any setting within ~12 hrs. of the correct time is acceptable, because Seaglider gets an accurate time from its first GPS fix and resets the internal clock.

   **Note:** The format for setting the date and time is: **mm/dd/yyyy hh:mm:ss** with no missing values.

9. When prompted, “Are you running on external (bench) power?,” do one of the following:
   - If you are running on internal Seaglider power, press ENTER to accept the default anser (**N** for No).
   - If you are running on external bench power, type **Y** (for Yes) and press ENTER.

   The Main Menu with five (5) numbered items displays (see Figure 3-14).

**FIGURE 3-14. Seaglider Set Date, Time, and Power Source**

```
version 66.07/18:775 starting up Mar 29 2011 10:36:24
Seaglider operating software developed and maintained by
Applied Physics Laboratory, University of Washington
Copyright 2003-2010, University of Washington
Built with SCENARIO, without ICE, without RAFOS, without KERMIT,
     without ARS, without MMODEM, without PAAM, without AQUADOPP, without SAILCT,
     without DEEPGLIDER, without HEAP_RECYCLE, with REV_C, without LUA
with serial and logger device sensor integration facilities
developed by Integrative Observational Platforms Group, APL-UW
_mtop = 0x2fefff, _mbot = 0x28cdb8, _mcur = 0x28e000
glmalloc_init: glmalloc_data_init = 0x0
glmalloc_init: avail_bytes = 434175, requested_bytes = 358400
glmalloc_init: Allocated 358400 bytes.
2.386,SSENSOR,N,reading cnf gpctddo.cnf
2.735,SSENSOR,N,assigned WL_BBFL2VMT to sensor slot 1 (p = 83)
2.907,SSENSOR,N,assigned GPCTD to logger sensor slot 1 (p = 85)
3.911,SPOWER,N,Iridium_during_init saved current=0.10300
4.080,SPOWER,N,Iridium_during_connect saved current=0.16000
4.252,SPOWER,N,Iridium_during_xfer saved current=0.22300
4.421,SPOWER,N,GPS saved current=0.02696
4.569,SPOWER,N,TT8 saved current=0.01496
4.716,SPOWER,N,TT8_Active saved current=0.01421
4.873,SPOWER,N,TT8_Sampling saved current=0.03743
5.033,SPOWER,N,TT8_CF8 saved current=0.04717
5.187,SPOWER,N,TT8_Kalman saved current=0.05915
5.347,SPOWER,N,Analog_Circuits saved current=0.01200
5.511,SPOWER,N,Compass saved current=0.01573
5.712,HCOMPASS,N,Active compass is SP3003
5.875,SSYS,N,Changing capture file
5.967,SSENSOR,N,sensor SBE_CT not found in installed sensors!
6.092,SSENSOR,N,sensor SailCTx not found in installed sensors!
6.212,SSENSOR,N,no CT sensor configured
9.538,SUSR,N,Glider 535, mission 0, dive 3, booted Thu Jan  1 00:00:09 1970
, boot #16
10.603,HXPDR,N,max range = 60.000000 m, timeout = 80 ms
10.901,SSURF,N,Hit CR within 1 minute to start, otherwise will go to recovery ...
Set current date and time
<CR> for default [03/29/2011 10:36:24]
Are you running on external (bench) power? [N]

------ Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 2
```

## Checking the Primary and Alternate Phone Numbers

To check the primary and alternate telephone numbers that Seaglider calls to connect to the basestation:

1. Type **1** for Parameters and Configuration, and then press ENTER.
2. Type **7**, and then press ENTER.
3. If the primary telephone number is absent or incorrect, enter the primary basestation telephone number (see Figure 3-15 on page 53).

   The phone number includes the country code, area code, exchange and number.

For example: In the US the country code is 1. If the primary number of the basestation in the United States is (919) 123-4567, input ‘19191234567’ into Seaglider.

In the UK the country code is 44. For example, if the telephone number of the basestation is (020) 1234 5678, input ‘4402012345678’ into Seaglider.

**Note:** DO NOT include 00 at the beginning of the phone number to signify an international call. Seaglider adds this automatically.

**FIGURE 3-15. Verify Primary Telephone Number**

```
------ Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 1                                   Step 1

------ Edit parameters ------
*Flight control and mission definition
     1 [basic   ]  Basic mission and glider parameters
     2 [dive    ]  Dive parameters
     3 [flight  ]  Flight parameters
     4 [surface ]  Surface parameters
     5 [rafos   ]  RAFOS parameters
     6 [password]  Set/show glider login password
     7 [telnum  ]  Set/show basestation phone number
     8 [altnum  ]  Set/show basestation alternate phone number
*Pitch, roll, VBD
     9 [pitch   ]  Pitch parameters
    10 [roll    ]  Roll parameters
    11 [vbd     ]  VBD parameters
*Sensors and peripherals
    12 [config  ]  Hardware configuration parameters
    13 [pressure]  Pressure (external) parameters
    14 [intpress]  Pressure (internal) parameters
    15 [compass ]  Compass paramaters
    16 [altim   ]  Altimetry parameters
    17 [seabird ]  Seabird CT calibration
    18 [power   ]  Power parameters
*Utility
    19 [all     ]  Edit all parameters
    20 [validate]  Validate parameters
    21 [details ]  Show parameter details
    22 [show    ]  Show changed parameters
    23 [clear   ]  Clear changed parameters
    24 [save    ]  Save parameters by name to file
    25 [dump    ]  Dump parameters to screen
    26 [load    ]  Load parameters from file
    27 [reset   ]  Reset to defaults
    CR)  Return to previous
Enter selection (1-27,CR): 7 ———————— Step 2
6561.080,SUSR,N,Current telnum is 12062215341 ———————— Step 3
New telnum (15 char max length, CR to leave unchanged): 19194841429
6605.038,SUSR,N,Changing telnum to 19194841429
```

4. Type **8**, and then press ENTER (see Figure 3-16 on page 54).

**FIGURE 3-16. Verify Alternate Telephone Number**

```
------ Edit parameters ------
*Flight control and mission definition
     1 [basic    ]  Basic mission and glider parameters
     2 [dive     ]  Dive parameters
     3 [flight   ]  Flight parameters
     4 [surface  ]  Surface parameters
     5 [rafos    ]  RAFOS parameters
     6 [password]   Set/show glider login password
     7 [telnum   ]  Set/show basestation phone number
     8 [altnum   ]  Set/show basestation alternate phone number
*Pitch, roll, VBD
     9 [pitch    ]  Pitch parameters
    10 [roll     ]  Roll parameters
    11 [vbd      ]  VBD parameters
*Sensors and peripherals
    12 [config   ]  Hardware configuration parameters
    13 [pressure]   Pressure (external) parameters
    14 [intpress]   Pressure (internal) parameters
    15 [compass ]   Compass paramaters
    16 [altim    ]  Altimetry parameters
    17 [seabird ]   Seabird CT calibration
    18 [power    ]  Power parameters
*Utility
    19 [all      ]  Edit all parameters
    20 [validate]   Validate parameters
    21 [details ]   Show parameter details
    22 [show     ]  Show changed parameters
    23 [clear    ]  Clear changed parameters
    24 [save     ]  Save parameters by name to file
    25 [dump     ]  Dump parameters to screen
    26 [load     ]  Load parameters from file
    27 [reset    ]  Reset to defaults
    CR)  Return to previous
Enter selection (1-27,CR): 8
6699.777,SUSR,N,Current altnum is 12062217301
New altnum (15 char max length, CR to leave unchanged): 19193612847
6725.312,SUSR,N,Changing altnum to 19193612847
```

5. If the alternate telephone number is absent or incorrect, do the following:
   - Enter the alternate basestation telephone number, if available.

     The method is the same as that for the primary telephone number. If no alternate telephone number is available, leave the entry blank.
   - After you have entered the last telephone number, press ENTER twice to exit to the Main Menu.

   For a complete listing of menu options available, when directly connected to Seaglider, and their function, see Appendix F, “Hardware and Configuration Menus” on page 367.

### Checking the Pump, Motors, and Basestation Communication

To check the proper operation of the VBD pump, pitch and roll motors and Iridium communication with the basestation:

1. Disconnect the communication cable from the laptop and the Seaglider.

2. Move Seaglider outside to an area where you can position it with the antenna pointing up, with an unobstructed view of the sky.

   **Note:** If moving Seaglider outside is easier without the antenna mast mounted to the aft end of Seaglider, detach it. See "Checking Out the Seaglider System" on page 47. DO NOT disconnect the antenna cable for this procedure. Once outside, reinstall the antenna, if necessary, following steps on page 42.

3. Connect the communication cable to Seaglider.

4. Position Seaglider, in its cradle, nose down, antenna pointing up within 40° of vertical and make sure the antenna has an unobstructed view of the sky.

   **Note:** Secure Seaglider so that it cannot fall over.

5. Connect the communication cable to the laptop and power up the laptop.

6. Start the terminal emulation program and open a capture file.

7. Turn on Seaglider and set the date, time and power following steps 5-9 on page 49 through page 52.

   The Main Menu with five (5) numbered items displays (see Figure 3-17).

8. Type **5** (for Pre-Launch), and then press ENTER (see Figure 3-17).

**FIGURE 3-17. Pre-Launch**

```
------ Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 5

------ Launch Menu ------
     1 [scene   ]  Set scenario mode
     2 [selftest]  Perform interactive self test
     3 [autotest]  Perform autonomous self test
     4 [uploadst]  Upload self-test results
     5 [reset   ]  Reset dive/run number
     6 [test    ]  Test Launch!
     7 [sea     ]  Sea Launch!
    CR)  Return to previous
Enter selection (1-7,CR): 3
```

9. Type **3** (for Perform Autonomous Self Test), and then press ENTER.

   See Appendix E, "Autonomous Self Test" on page 347 for sample output from an autonomous self test.

   **Note:** See "Interactive Pre-launch Self Test" on page 188 for information on the interactive self test.

**Note:** You do not have to do anything during the self test, but you should monitor the process on the laptop for any warnings or errors.

An example of the files sent to the basestation via Iridium at the end of the self test as well as the output after the files are processed on the basestation are listed below. Carefully review the processed files for any warnings or errors.

In the following example files (generated by Seaglider during a self test and sent to the basestation via Iridium), 'st' stands for self test and '0007' means this is the seventh self test done on this Seaglider.

st0007du.b.1a.x00

st0007du.b.x00

st0007du.r

st0007du.x00

st0007du.x00

st0007du.x00

st0007kz.r

st0007kz.x

The basestation processes the output files and produces the files listed below. The 'pt' portion of the file name stands for processed test. The next three digits are Seaglider's number and the 0007 is the number of the self test. See "Seaglider-Generated Files" on page 246 for an explanation of each file.

pt5130007.asc

pt5130007.cap

pt5130007.dat

pt5130007.eng

pt5130007.log

pt5130007.asc

**10.** Do one of the following:

| IF... | THEN... |
|---|---|
| There are no warnings or errors in any of the files (expected outcome) the test is complete. | Go step 11. |
| There is a problem during the testing, the final lines from the self test are: "Self test FAILED or ABORTED!" | Review the self test to find the reason:<br>• If there is a problem with the pitch, roll or VBD, contact iRobot customer service.<br>• If there is a problem with the gps or with the Iridium connection or transfer of files over Iridium, reposition Seaglider by changing its location and/or its angle toward the sky, and then re-run the automated self test.<br>• If , after several auomated self test iterations, there are still errors with the gps, Iridium, or file transfer, contact iRobot customer service. |

**11.** Turn off Seaglider by moving the magnetic wand over the OFF symbol on the port side of Seaglider 20" aft of the nose for at least 1second while simultaneously pressing ENTER (or have someone else press ENTER) on the laptop keyboard. Continue moving the wand around the OFF symbol and pressing ENTER until the laptop screen shows no response to pressing ENTER.

**Note:** Make sure that the wand is touching the fairing.

**12.** Disconnect the communication cable from Seaglider and the laptop and reinstall the dummy plug on Seaglider communication connector.

**13.** Detach the antenna mast from the aft fairing before storing or moving Seaglider in its cradle. See step 17 on page 46.

For information on stowing and moving Seaglider, see "Transporting Seaglider to the Field" on page 174.

CHAPTER 4

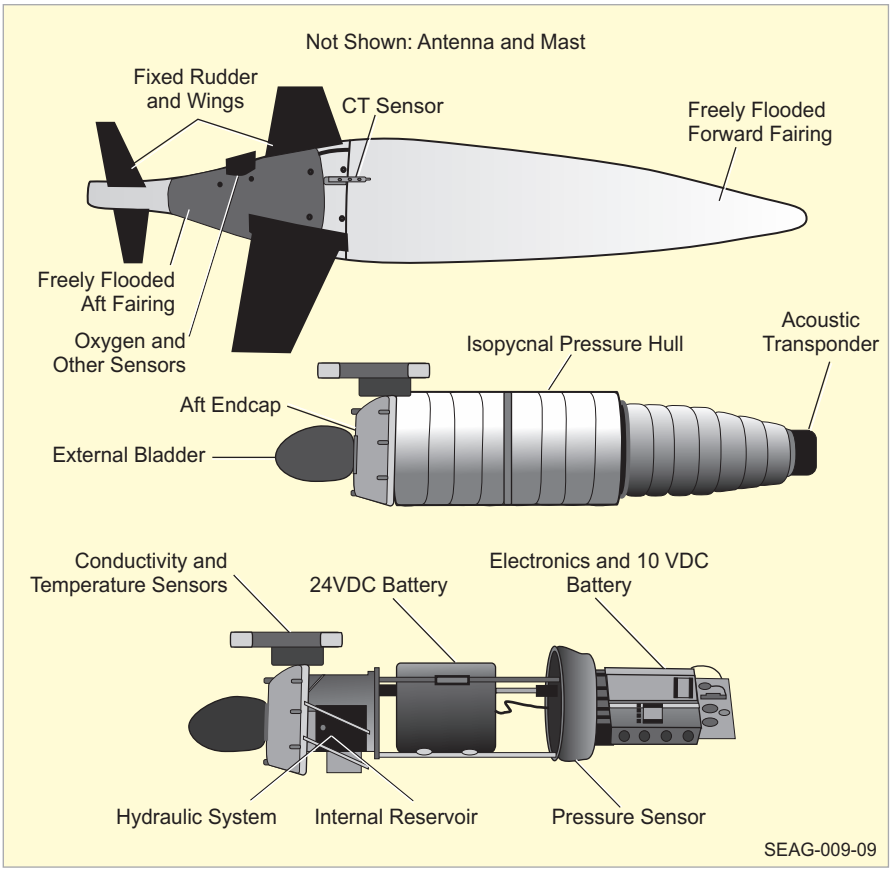
# *Operating Principles*

This chapter details the how to operate the 1KA Seaglider. The following topics are covered:


- "Seaglider Components" on page 60
- "Principles of Seaglider Operation" on page 65
- "Control of the Static Forces" on page 69
- "Features of Control" on page 73


Unlike an airplane, there are no moving external control surfaces on Seaglider. To understand how Seaglider moves through the water in a sawtooth pattern, see Figure 4-1 on page 60 and the descriptions of Seaglider's primary parts and how they control Seaglider's movement. Your configuration may differ, depending on what you ordered.

**FIGURE 4-1. iRobot 1KA Seaglider and Its Primary Parts**




## *Seaglider Components*

### Isopycnal Pressure Hull

An important and unique feature of Seaglider is the compressibility of its pressure hull. For maximum efficiency, Seaglider uses an isopycnal pressure hull, which is designed to have the same compressibility as seawater.

Other gliders have rigid pressure hulls that are designed to maintain a fixed volume at all rated pressures. As gliders with non-isopycnal hulls dive, any density increase

in the surrounding water causes them to acquire positive buoyancy. This requires compensation (subtraction of displaced volume) to maintain a constant buoyancy difference. That same compensation has to be recovered by pumping to achieve positive buoyancy when Seaglider ascends.

Seaglider's isopycnal hull eliminates that need, as the pressure hull does not acquire positive buoyancy from the compression of the surrounding seawater. For dives to 1000 m, this results in about a 10% energy savings in the 24V energy budget.

### Inside the Pressure Hull

The following components are located inside the isopycnal pressure hull:

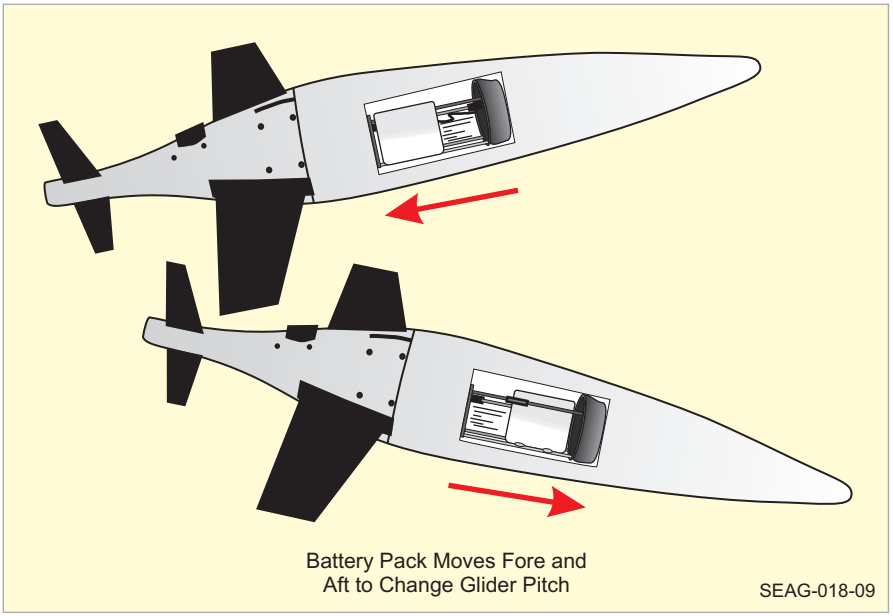
- Mass shifter with the 24 VDC lithium primary battery
- Main electronics board assembly with the 10 VDC lithium primary battery attached underneath
- Internal components of the hydraulic system

**Caution:** The components inside the pressure hull are **NOT** user serviceable. Any unauthorized opening of the pressure hull voids the one year warranty. (See Appendix J, "Warranty and Disclaimers" on page 441.)

#### *Mass Shifter with 24 VDC Battery Pack*

The 24 VDC lithium primary battery pack can be moved by the mass shifter mechanism forward and aft to control vehicle pitch and side to side to control vehicle roll. (See Figure 4-2 on page 62.)

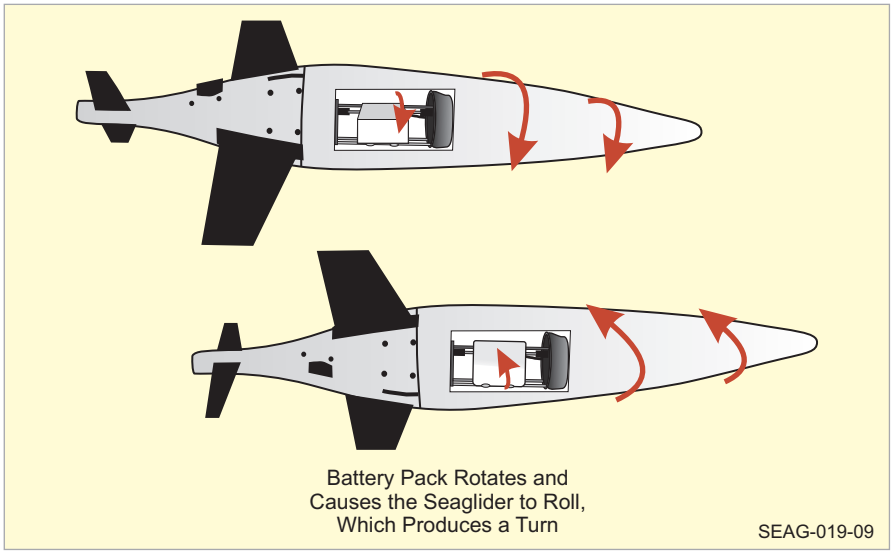
**FIGURE 4-2. Mass Shifter Causing Pitch Change**




Aside from providing the weight needed to change the pitch of the vehicle, the approximately 9000g battery pack also powers Seaglider's pumps, motors, communications, and transponder.

A brass weight is attached to the bottom of the 24 VDC battery pack. This weight provides the axial asymmetry necessary to make Seaglider roll when the mass shifter is rolled to the port or starboard side. (See Figure 4-3.)

**FIGURE 4-3. Mass Shifter Causing Roll Change**




### *Main Electronics Assembly with 10VDC Battery Pack*

The microprocessors and electronics that control Seaglider's flight (acoustic transponder, pitch, roll, buoyancy communications, GPS) and science sensors are located on the main electronics board. A 10 VDC lithium primary battery pack is installed underneath the electronics assembly. This battery, which weighs approximately 2600g, powers the pressure and science sensors, processor, GPS, transponder, and main electronics.

## Outside the Pressure Hull

The acoustic transponder, fairing, rudder, wings, antenna and science sensors are located outside of the isopycnal pressure hull. The following sections describe these components.

### *Acoustic Transponder*

The transponder, located on the front end of the pressure housing, is used for two separate functions:

- Seaglider location during recovery
- Altimetry during dives

### *External Fairing*

The outer shell, which includes the forward and aft fairings, allows Seaglider to move through the water smoothly, cutting down on drag and providing maximum energy efficiency.

### *Rudder and Wings*

The rudder and wings are fixed. The wings (with a combined span of 1m) produce lift vectors, relative to CG (Center of Gravity) and CB (Center of Buoyancy), that turn Seaglider when rolled and cause the glider to move horizontally when net buoyancy is positive or negative and vehicle pitch is not past the stall angle. The rudder provides yaw stability as Seaglider moves forward.

### *Antenna Mast*

The antenna mast is attached to the aft end of Seaglider. It provides height to the antenna that is located at the top of the mast.

The antenna serves both the GPS and the Iridium modem:

- When Seaglider first surfaces, the antenna is switched to the GPS, allowing Seaglider to get a latitude/longitude position.
- The antenna is then switched to the Iridium modem, allowing Seaglider to:
  - Call the basestation
  - Upload data files, including its current position, to the basestation
  - Download new command, target, science and pdos files from the basestation
- Once the communication session with the basestation is complete, the antenna is switched back to the GPS and a GPS fix is obtained before beginning the next dive.

### *Science Sensors*

Seaglider comes standard with a Sea-Bird Electronics Conductivity and Temperature (CT) sensor. You can purchase additional sensors for installation on Seaglider. Other sensors that customers have had installed are WET Labs ECO Pucks, PAR, GPCTD, Aanderaa, and dissolved oxygen sensors (pumped) by Sea-Bird.

### Inside and Outside the Pressure Hull

The Variable Buoyancy Device (VBD) is located both inside and outside of the isopycnal pressure hull. The VBD is a hydraulic system that achieves a specified total vehicle displacement by varying the size of an oil-filled bladder external to the pressure hull.

| To make the vehicle... | The VBD... |
|---|---|
| more buoyant | pumps oil from a reservoir located inside the pressure hull into the bladder to increase the displacement of the vehicle. |
| less buoyant | bleeds oil from the bladder back into the reservoir to decrease the vehicle's displacement. |

See "Buoyancy" on page 67 for more information.

## *Principles of Seaglider Operation*

There are a number of factors that affect how a Seaglider operates, including the water density and currents in the mission area and the static and dynamic forces acting on Seaglider. The following sections describe these factors, and how Seaglider is designed to deal with them.

### Environmental Factors

#### *Density*

Density is defined as mass per unit volume: $\rho=m/V$. Dimensionally, this is equivalent to:

$$\rho=m/L$$

Oceanographers routinely switch between SI(mks) and cgs units when referring to seawater densities. Densities are specified in $g/cm^3$ (with a typical 1000m ocean value of $1.0275 g/cm^3$ relative to sea surface pressure) or $kg/m^3$ (with a typical 1000m ocean value of $1027.5 kg/m^3$). Oceanographers have a shorthand notation for density, called $\sigma$, which is defined as:

$$\sigma = (\rho-1000)kg/m^3$$

Thus, the typical 1000m ocean water density in σ units is 27.5. In addition, we use the unit $\sigma_T$, which is defined as σ of a sample of water at a specific temperaturre and salinity at standard atmospheric pressure. Densities discussed in Seaglider operations are typically given in cgs units ($g/cm^3$).

### *Stratification*

Stratification is the term used to describe the density layering of the ocean, with denser water below lighter water (stable stratification). Strong stratification means a large change in density between two depths while weak stratification is a small change in density between two depths.

### *Currents*

The depth-averaged current over the course of a dive influences the distance covered over the ground (DOG) by Seaglider. The depth-averaged aspect is important. Seaglider can make progress towards a waypoint even in the presence of strong adverse surface currents by diving through deeper waters with more favorable currents. The maximum depth-averaged current that Seaglider can stem is 40cm/s, or 0.8kts. These dives tend to be done with large negative thrust on the dive (-350cc), and vertical velocities of 18cm/s. The dives take about three hours between surfacing, or about eight dives per day in 1000m of water.

It has been shown in deployments in the Kuroshio that Seaglider can make crossings of a strong western-boundary current. This is typically done in a triangular track, with an inshore and then an offshore transect of the strong current, followed by a return upstream in the calmer water offshore of the strong current.

One might imagine interesting tracks in the equatorial Pacific that would exploit the equatorial undercurrent. Sharp vertical shear in the currents can induce turning moments on Seaglider's body. Large vertical velocities (upwelling or downwelling) can introduce large changes in vertical velocity, and in some cases cause dives to truncate or abort prematurely.

For information on strategies for dealing with the environmental conditions, see Chapter 6, "Pre-Deployment Tasks" on page 159.

## Static Forces

Seaglider's flight is controlled by systems that change buoyancy, pitch and roll. It is designed to operate within several hundred cubic centimeters (cc) of neutral buoyancy over a seawater density range of $10\sigma_T$.

- Buoyancy is controlled by changing the displaced volume of Seaglider.
- Pitch is controlled to put Seaglider in a nose up position for climbing, and a nose down position for diving and exposing the antenna at the surface. Pitch is controlled by altering the center of mass of the vehicle by moving the battery mass forward or aft.
- Roll is controlled to cause Seaglider to turn. Roll is changed by altering the center of mass of the vehicle by rotating the battery mass from side to side.

### Gravity

Gravity is the force that pulls objects toward the center of the planet. The center of gravity of Seaglider is changed by the movement of the mass shifter inside Seaglider.

The mass shifter is moved forward and back to effect changes in the vehicle's pitch and from side to side to effect vehicle roll.

Seaglider achieves static trim by the addition of ballast weight between the fairing and the pressure hull. The position and amount of ballast is determined by mission and trim requirements. The addition or removal of science sensors also affects the required ballast weight.

### Buoyancy

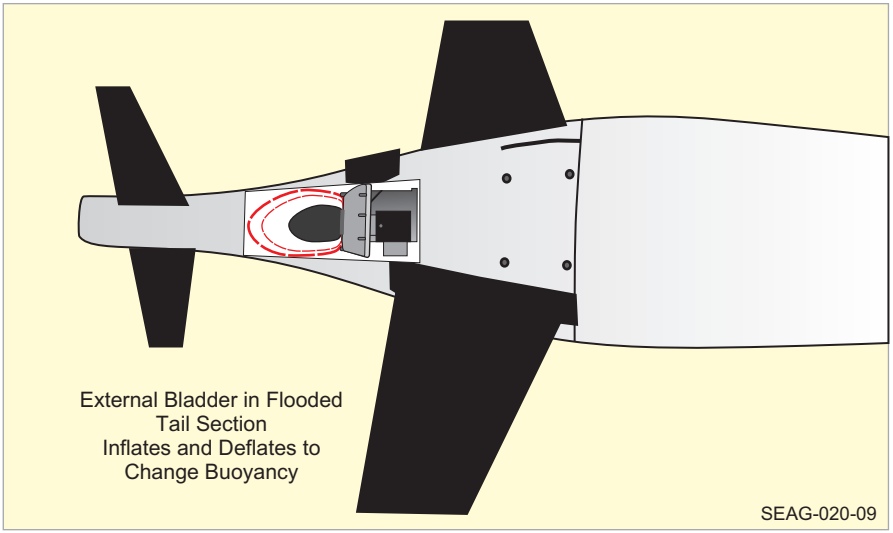
Buoyancy is the unbalanced (positive or negative) vertical force on a submerged object arising from the vertical pressure gradient. It was Archimedes who, as mentioned previously, stated that the buoyant (upward) force on a submerged object is equal to the weight of the fluid that is displaced by the object. The buoyancy of a submerged object is altered by changing its density, either by changing its mass or volume.

Submarines typically alter their buoyancy by changing their mass while maintaining their volume. Seagliders change their buoyancy by changing their displaced volume while keeping their total mass fixed. This is done by moving hydraulic oil between the reservoir inside the pressure hull and the bladder external to the pressure hull:

- Pumping oil from the internal reservoir to the external bladder increases Seaglider's displaced volume which increases its buoyancy.
- Bleeding oil from the external bladder into the reservoir decreases Seaglider's displaced volume which decreases its buoyancy.

Figure 4-4 shows the location of the bladder.

**FIGURE 4-4. External Bladder Inflation and Deflation**




## Dynamic Forces

### *Lift*

Seaglider gets lift from its body and wings, which convert the vertical force provided by the variable buoyancy device (VBD) into horizontal motion. Some additional lift comes from the rudder (vertical stabilizer) while banked (executing turns).

### *Drag*

The shape of Seaglider's hull was designed to maintain laminar flow over 70% of the hull length (between nose and CT sail) [Eriksen et al, 2001, Humphreys, Smith, et al., 2003]. Drag is partitioned into two types in Seaglider's flight model:

- Induced drag
- All other types of drag (for example, skin friction, form drag)

Drag is caused by anything projecting from Seaglider. Drag produced by sensor protrusion creates a large portion of total vehicle drag. The CT (Conductivity and

Temperature) sensor on top, for example, causes approximately 30% of the vehicle's drag.

Drag is proportional to the square of speed, so reducing vehicle speed will increase the efficiency of thrust used relative to distance traveled.

### Hydrodynamic Model

A hydrodynamic model for Seaglider is used by pilots to help with buoyancy trim and is used to evaluate depth-averaged currents.

The model has three parameters, traditionally called a, b and c:

- Lift
- Drag
- Induced drag

For our purposes, it is convenient to think of the hydrodynamic model as a black box that produces estimates of Seaglider's velocity as a function of computed buoyancy, observed pitch, and water density:

$$v_{model} = F(buoyancy_{computed}, pitch_{observed,} \text{ water density}).$$

The $v_{model}$ can be resolved into horizontal and vertical components. In particular, the horizontal component, $u_{model}$, can be used with the observed compass headings throughout a dive to determine a dead-reckoned Seaglider track through the water. This results in a predicted surfacing position, based on the GPS determined dive starting point. The difference between this predicted surfacing position and the actual GPS determined surfacing position is what provides the estimate for depth-averaged current. Similarly, the vertical component, $w_{model}$, can be compared with $w_{observed} = dp/dt$, to adjust the VBD trim and then to estimate vertical velocities in the water column.

## *Control of the Static Forces*

During Seaglider operations, the pilot must monitor and control the vehicle pitch, roll, and buoyancy. Positions monitoring systems controlling these three things are encoded by potentiometers, digitized by 4096-count analog-to-digital (A/D) converters.

The A/D counts run from 0 to 4095. Physically attainable limits (also called hardware limits) for each system are determined empirically at the time of assembly. A safety margin is added to these physical limits to arrive at software limits, which are the positions (in A/D counts) beyond which Seaglider's operating software will not command that particular system.

Associated with each system are the following:

- A center position, which is intended to be the vehicle neutral for that system, in a particular environment.
- A factor that converts A/D counts to physical displacement, based on the mechanical design.
- A gain that relates movement of each system to the effect it has on the Seaglider.

### Pitch

Pitch is controlled by moving the 24V battery pack forward and aft along the longitudinal axis of Seaglider. (See Figure 4-1 on page 60.) The motion is accomplished by an electric motor, geared to drive a worm-gear in such a way that 319.92 A/D counts equals 1 cm of battery mass travel (**$PITCH_CNV**).

Seagliders typically respond to movement of the battery pack in the longitudinal axis by pitching 25-32° per centimeter of mass travel. This **$PITCH_GAIN** is a parameter, as it is dependent on the particular sensor suite and trim ballast installed on each Seaglider.

The following are some typical pitch ranges and values for Seaglider.

**TABLE 4-1. Typical Pitch Ranges and Values**

| | **Hardware Limit (A/D Counts)** | **Software Limit (A/D Counts)** |
|---|---|---|
| **Full forward (nose down, -)** | 20 | 70 (**$PITCH_MIN**) |
| **Full aft (nose up, +)** | 3402 | 3352 (**$PITCH_MAX**) |
| **$C_PITCH** ***(example)*** | | 2346 |

Pitch movement (cm) = (Pitch Observed(counts)-**$C_PITCH**(counts)) x **$PITCH_CNV** (cm/(counts))

Where:

**$C_PITCH** = pitch center position

**$PITCH_CNV** = pitch position conversion factor

While A/D counts are always positive, displacement can be positive or negative, relative to a given **$C_PITCH**. Pitch is usually trimmed so as to have 70% of the pitch travel available for pitching down (forward of **$C_PITCH**), and 30% available for pitching up (aft of **$C_PITCH**).This is to ensure a good surface position with Seaglider sufficiently pitched down to fully expose the antenna.

### Roll

Roll is controlled by rotating the 24V battery pack inside the hull. The pack is axially asymmetric and weighted (1100g brass weight) on its ventral face (as normally installed). An electric motor and gear train rotate the mass such that 35.37 A/D counts is equivalent to 1 degree of battery mass rotation (**$ROLL_CNV**).

Seagliders typically respond to the rotation of the battery pack by rolling 1/2º for every 1º of battery pack rotation. The response to the battery pack rotation is also dependent on the amount and distribution of trim lead.

The control strategy is to roll the 24V battery pack a specified amount (40º) in the appropriate direction when a turn is initiated, and then roll back to neutral (center) when the correct heading is reached. (See Figure 4-3 on page 63.)

Setting **$ROLL_GAIN_P** greater than 0 will enable the proportional roll controller. This algorithm allows for smaller roll mechanism movements which yield lower power use and finer heading control. The following equation shows how **$ROLL_GAIN_P** is used by the proportional roll controller during a dive to figure out how much to move the roll mechanism.

Roll = (Desired Head - Actual Head) * **$ROLL_GAIN_P**

Clipped to +/- **$ROLL_DEG**

Typical values used for **$ROLL_GAIN_P** are 0.5 or 1.0.

**Note:** Seaglider turns in the opposite sense from its bank angle on the dive (opposite from upright airplane control), and in the same sense as its bank angle on the climb (same as upright airplane control).

Table 4-2 provides some typical roll ranges and values for Seaglider. Two roll centers, dive and climb, are used because asymmetries in Seaglider form (where components are installed inside Seaglider) result in different roll trim on dives and climbs.

**TABLE 4-2. Typical Roll Ranges and Values**

| | Hardware Limit (A/D Counts) | Software Limit (A/D Counts) |
|---|---|---|
| Full Roll to Port (-) | 0 | 150 (**$ROLL_MIN**) |
| Full Roll to Starboard (+) | 3983 | 3833 (**$ROLL_MAX**) |
| **$C_ROLL_DIVE** ***(example)*** | | 2000 |
| **$C_ROLL_CLIMB** ***(example)*** | | 2050 |

## Buoyancy

Buoyancy is controlled by a mechanism called the Variable Buoyancy Device (VBD). It is a hydraulic system whose purpose is to maintain a specified total vehicle displacement by varying the size of an oil-filled bladder external to Seaglider's pressure hull. The system pumps oil from an internal reservoir into the external bladder to increase displacement and allows oil to bleed from the external bladder into the internal reservoir to decrease displacement (see Figure 4-4 on page 68).

Linear potentiometers on either side of the internal reservoir measure the position of the reservoir's rolling diaphragm. The mean of the two values is reported as the position of the diaphragm, which can be interpreted as the amount of oil in the internal (or external) reservoir. The geometry of the system results in 4.0767 A/D counts per $cm^3$ of oil (**$VBD_CNV**). The point of neutral buoyancy is designated **$C_VBD**, and is set relative to the densest water to be encountered on a mission.

VBD control is calculated to achieve specific results, which depend on the pilot specified quantities:

- Seaglider vertical velocity
- Distance to next waypoint
- Maximum glide slope
- Rho (density)

VBD control is the “gas pedal” or throttle that controls vehicle horizontal velocity. Specific VBD control issues are discussed in more detail in Chapter 6, “Pre-Deployment Tasks” on page 159.

Typical VBD ranges and values for Seaglider are described in Table 4-3.

**TABLE 4-3. Typical VBD Ranges and Values**

| | Hardware Limit (A/D counts) | Software Limit (A/D/count) | Volume ($cm^3$) |
|---|---|---|---|
| $V_{max}$ | 105 | 205 (**$VBD_MIN**) | 557 (with respect to **$C_VBD**) |
| $V_{min}$ | 3610 | 3510 (**$VBD_MAX**) | -266 (with respect to **$C_VBD**) |
| Range | | | 823 |
| **$C_VBD** | | 2476 | |

**Note:** Vmax is the maximum displaced volume of Seaglider, and Vmin is the minimum displaced volume of Seaglider. When given in $cm^3$, they are with respect to a given **$C_VBD**.

## *Features of Control*

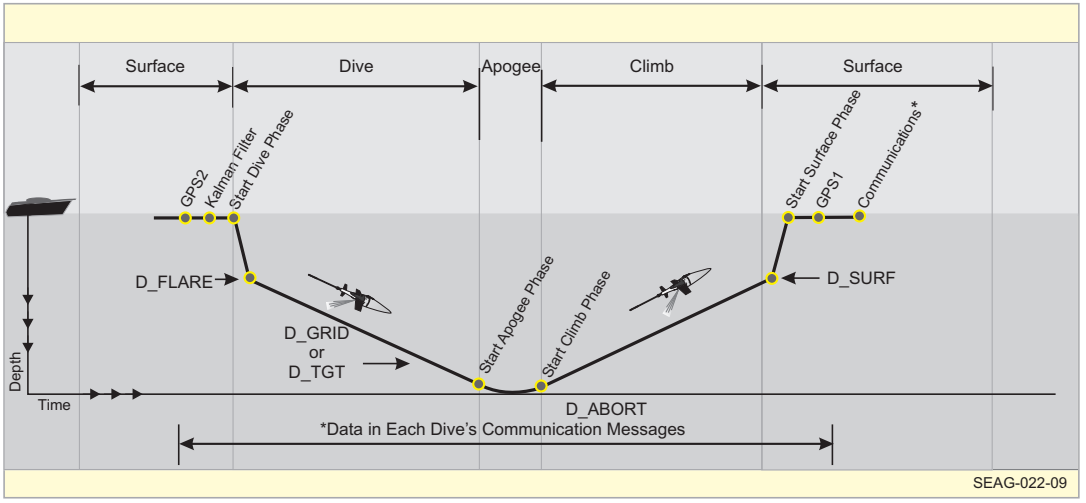
### Canonical Dive

Seaglider performs its mission by repeating a canonical dive until either it is commanded to stop or until an abort condition is reached. Numerous aspects of the canonical dive are under the control of the pilot through an extensive set of parameters. A few are indicated in Figure 4-5. Many more are not shown, but explanations for them can be found in Chapter 5, “Piloting Parameters” on page 83.

The run phases of the dive are indicated by the intervals at the top of Figure 4-5 and the profile data boundaries are indicated by the interval at the bottom of the figure.

**Note:** Figure 4-5 is not to scale in either dimension.

**FIGURE 4-5. Canonical Dive Schematic**




## Control Design

Seaglider's flight control scheme has two guiding principles:

- Maintain constant vertical velocity during a dive
- Minimize the total energy expenditure during a dive

Constant vertical velocity is desired because Seaglider samples its sensors evenly in time. Constant vertical velocity then implies that the samples are equally spaced in depth. Sample intervals are specified by the pilot through the science file. The intervals can vary by pilot specified depth bands, but are uniform within each specified depth band.

The vertical velocity is not specified directly by a parameter, but is calculated from parameters that describe:

- Target depth of a dive (**$D_TGT**), in meters
- Time to complete a dive (**$T_DIVE**) from surface to surface, discounting pumping time at the bottom of the dive, in minutes

Therefore, the vertical velocity, in cm/sec, is:

$w_d = (2 * \mathbf{\$D\_TGT} * 100\text{cm/m})/(\mathbf{\$T\_DIVE} * 60\text{s/min})$

Seaglider operating software chooses the buoyancy and pitch used on any individual dive to achieve the best results on that dive. The choices are bounded by the parameters:

- **$MAX_BUOY**: the maximum negative buoyancy allowed on a dive
- **$GLIDE_SLOPE**: the maximum glide slope allowed on the dive

The choices are also bounded by physical limits, neutral buoyancy (need some negative buoyancy to glide) and the stall angle. The software has to choose a buoyancy value between 0 (neutral) and **$MAX_BUOY**, and a desired pitch angle between the stall angle and **$GLIDE_SLOPE**.

The choice is determined by the distance to the next waypoint. The pitch angle is chosen to achieve the desired horizontal distance: maximum pitch if the waypoint is close, minimum pitch if the waypoint is distant, or the exact distance, if possible. Once the pitch angle is chosen, the buoyancy is chosen to achieve the desired vertical velocity in the densest (deepest) water.

The main energy draw on Seaglider is pumping hydraulic oil from the internal reservoir to the external bladder at depth, where the pump has to overcome the seawater pressure acting on the bladder. Since the pump can consume as much as 70% of the energy budget of Seaglider, control during flight is generally designed to minimize the total amount of pumping required on a dive. In particular, no additional bleeding post initial bleed is allowed on descent (dive) to maintain the vertical velocity. Pumping as necessary is allowed on the climb to maintain the vertical velocity.

Pitch is essentially steady during each phase of the operation, with the exception of slight pitch maneuvers on the climb to compensate for the changes in mass distribution and buoyancy due to pumping oil from the internal reservoir into the bladder. See “Run Phases” on page 77 for detailed information on the control scheme.

Seaglider can cover up to 20 km/day through the water and can station-keep within a factor of two of the dive depth (2 km horizontal distance on 1 km vertical distance dives, 200 m on 100 m dives). The navigation system on Seaglider is waypoint-based, not track-based meaning the system decides on the most efficient way to reach the next waypoint, but does not attempt to stay on a given track. Track-based navigation can be approximated by using more waypoints along a track.

### Sampling

Sensor sampling intervals are specified in the science file. The practical lower limit on sampling is 4 seconds if only the conductivity and temperature sensors are sampled, but with additional sensors installed, 5 seconds is the lower limit. The science file also gives the ability to turn off sensors, or only energize them every *nth* sample of other sensors, in a given depth range (or ranges).

### Sensor Interfaces

Prior to firmware version 66.07 science sensors (SBE CT, Aanderaa DO, SBE DO, WL BB2FL, WL BBFL2 etc.) were integrated by creating a driver that had to be written in the source code to accommodate the new sensor.

Starting with firmware 66.07, a new serial sensor interface, the Autonomous Logger Interface (ALI) became available. This interface, which can only be used for sensors with the ability to log data to an internal file system is .cnf configurable meaning that a new binary driver does not have to be written nor does the source code have to be modified each time a new type of smart sensor is added. The Sea-Bird pumped CTD (GPCTD – Glider Pumped CTD) is an example of a smart sensor connected to an ALI.

Another sensor interface named Ordinary Serial Interface (OSI) is also used to incorporate streaming serial sensors into the glider through the use of .cnf files.

### Bathymetry

Seaglider can read a digitized bathymetry map to determine how deep to dive, or can rely on the on-board altimeter to find the bottom and initiate the apogee maneuver. Bathymetry maps show the sea floor terrain as contour lines with associated depths.

Seaglider is less efficient operating in shallow water and most efficient in deep (up to 1000m) water. The practical shallow water limit is 50m. It is hard to make progress toward a waypoint in water shallower than that, for several reasons:

- Turn radius
- Pump time
- Surface time
- Currents

Seaglider's turning radius (a few tens of meters at typical 25 cm/s horizontal speeds) is such that a significant portion of a shallow-water dive can be spent

turning onto the correct course. Seaglider's pump is optimized for efficiency at pressures equivalent to 1000 m ocean depth. If an enhanced buoyancy engine (EBE) is installed, the glider is also optimized for pumping efficiency at 120 meters and shallower. Its rate at shallow-water pressures (2cc/s at pressures below 10 psi >2.3 cc/sec at pressures below 200 psi with the EBE); ~1.2cc/s at pressures greater than 10 psi) means that a significant portion of a shallow-water dive can be spent pumping.

Finally, time on the surface can be a significant percentage of the dive time. If surface currents or winds are adverse, Seaglider can lose as much or more distance toward a waypoint while on the surface as it gains on the dive.

Our operating guidelines are to operate in water deeper than 200m on offshore (deepwater) missions, and to try to stay in water deeper than 75m on coastal or estuarine missions. Seaglider is rated to 1000m, but the maximum depth that should be used for deep dives (**$D_TGT**) is 990m to allow for the apogee manuever.

### Run Phases

Launch and recovery phases are performed at the beginning and end of the mission. Surface, dive, apogee, and climb phases are meant to be repeated sequentially, once per profile, until the end of the mission. During the surface phase, GPS positions are acquired, communication with the basestation is accomplished, and navigation calculations for the next dive are made. Depth, time and functional triggers exist to cause Seaglider to move from one phase to the next.

Data acquisition is done in the dive, apogee, and climb phases of an autonomous run. During each of these phases, Seaglider collects data from the scientific instrumentation at a rate specified in the science file. Although other actions are performed during these phases, the data collection process is never interrupted. Another periodic action performed during the profile phases (dive, apogee and climb) is guidance and control (G&C). G&C operations occur at intervals defined in the science file and are done only when necessary.

The three G&C operations that can occur are:

- Pitch adjustment
- VBD adjustment
- Roll adjustment

When G&C operations occur, Seaglider is said to be in “active” guidance and control mode. When G&C corrections are not being made, Seaglider is said to be in “passive” G&C mode. These definitions of active and passive modes refer to G&C operations only. They do not apply to data acquisition intervals or activities. Seaglider is acquiring data during all profile phases, whether in active or passive G&C mode. In passive G&C mode, Seaglider’s processor enters a low-power sleep state between data acquisition points. Seaglider flies in the state specified in the previous active G&C mode.

### Launch

The launch phase begins when the field operator has initiated the Sea Launch procedure and all launch dialogue has completed. See Chapter 6, “Pre-Deployment Tasks” on page 159 for detailed information on the launch procedure. At launch, Seaglider is in its surface position (rolled to neutral, pitched fully forward and pumped to **$SM_CC** which is typically the maximum VBD for launch and enters a normal surface phase – acquiring GPS1 fix and initiating a communication session with the basestation via Iridium satellite.

### Surface

The surface phase begins at the end of the climb phase data acquisition. During the surface phase the following steps occur.

1. Surface Maneuver

   The surface position of Seaglider is pitched fully forward (to the software limit), rolled to neutral (**$C_ROLL_CLIMB**), and pumped to VBD = **$SM_CC**.

   If Seaglider surfaces with VBD > **$SM_CC**, no bleeding is done to force VBD = **$SM_CC**. There are several ways to enter the surface maneuver. Seaglider is in the surface position at launch, after normal completion of a dive (reached **$D_SURF**), in recovery phase, or after **$T_MISSION** minutes have elapsed from the start of the dive without achieving **$D_SURF** in climb phase.

   The first test in the surface phase is to check whether Seaglider’s depth is less than **$D_SURF**.

   - If it is, Seaglider pitches fully forward and pumps to **$SM_CC**.
   - If it is not, Seaglider first pumps VBD to its maximum value, and checks the depth again. If the depth is less than **$D_SURF**, Seaglider moves the pitch mass to its full forward position. This behavior is designed to try to get Seaglider to the surface in the event of a **$T_MISSION** timeout.

2. GPS1

   Once the surface position is attained, the GPS receiver is turned on, and left on, until a satisfactory position is acquired or until **$T_GPS** minutes have elapsed. This GPS position is called **$GPS1.** When this initial position is acquired, Seaglider waits an additional **$N_GPS** samples for a GPS position with an HDOP < 2.0 at which point acquisition stops and that position is accepted. If a fix with an acceptable HDOP is not received in **$N_GPS** samples, the last position is accepted and the size of HDOP is recorded in Seaglider's log files.

3. Communications

   Wireless communication via Iridium satellite begins following acquisition (or time out) of **$GPS1**. Seaglider powers up the Iridium phone, waits a specified time for registration with the Iridium system, and then attempts a data call to the basestation.

   Once the connection is established, Seaglider logs into the basestation as a dial-up user, and uses a modified XMODEM protocol to transfer files. Data and log files are transferred from Seaglider to the basestation, and command, control, diagnostic and special purpose files are transferred from the basestation to Seaglider. See Chapter 9, "Files for Operations" on page 237.

   If all file transfers were not accomplished, Seaglider waits **$CALL_WAIT** seconds and tries again. It tries to call up to **$CALL_TRIES** times, and if unsuccessful, continues with the surface phase, marking files as appropriate for later transfer, and incrementing the **$N_NOCOMM** parameter.

4. Measure Surface Depth and Angle

   After the communications session, Seaglider computes the average of 10 pressure readings, and then the average of 10 pitch angles to obtain a measurement of Seaglider's surface position. These values are written into the log file for the next dive.

5. GPS2

   After the surface pressure and pitch angle averages are completed, a second GPS fix, **$GPS2**, is acquired. This fix is the most recent position of Seaglider prior to diving.

6. Navigation and Flight Calculations

   The final component of the surface phase is the calculation of the parameters to determine Seaglider's flight path during the next profile: buoyancy, pitch angle and heading. These computations include the Kalman filter, if enabled, and the digital bathymetry table lookup, if enabled. Upon completion of the calculations, the surface phase is finished and a new dive phase (and new profile) is started.

### Dive Phase

The dive phase begins upon completion of the navigation and flight calculations that conclude the surface phase. Initially, pitch is in the full forward position and the VBD volume is equal to the endpoint of the surface maneuver. At the start of the dive phase, a VBD adjustment (bleed) is executed during the first guidance and control (G&C) operation to get Seaglider off the surface as quickly and with as much vertical velocity as possible. (Recall that pitch is still in the maximum forward position.) When Seaglider reaches a prescribed depth, **$D_FLARE**, it goes into a regular G&C operation (pitch, VBD, roll) to move to the desired pitch, VBD position, and course computed for the profile.

If Seaglider's speed is too fast on the dive section of the profile (too heavy), VBD pumping is not allowed to correct the speed error to conserve energy. As Seaglider descends into denser water, it becomes less negatively buoyant and slows down. If corrective pumping were allowed on the dive, it is possible that additional bleeding would be required to compensate as Seaglider reached denser water. That would then mean more pumping to eventually reach the buoyancy endpoint of the surface maneuver. Excess speed is tolerated on the dive to help minimize the total energy expenditure on the profile.

In the dive phase, Seaglider turns to starboard by banking to port and vice versa (opposite to upright aircraft flight).

### Apogee

When the target depth is reached, Seaglider enters the apogee phase. The apogee phase is a two G&C cycle procedure to smoothly transition from the dive phase to the climb phase without stalling.

During the first G&C cycle of this phase, Seaglider is pitched to an intermediate angle, **$APOGEE_PITCH**, rolled to neutral, and the VBD is pumped to 0 cc. The course adjustment and passive G&C mode are skipped.

A second G&C cycle is then executed and Seaglider is first pitched, then VBD is pumped, both to the inverse positions of the dive (pitch = -pitch, VBD = -VBD).

Data sampling continues throughout the apogee phase.

### Climb

The climb phase begins at the completion of the second G&C cycle of the apogee phase. Seaglider is positively buoyant and pitched up, headed for the surface at the same target vertical rate as achieved on the dive phase of the profile.

As in the dive phase, data acquisition and G&C continue at the intervals specified in the science file.

If Seaglider's speed is too fast on the climb section of the profile (Seaglider is too light), VBD bleeding is NOT allowed to correct the speed error to conserve energy.

There are two reasons for this methodology. First, any oil that is bled needs to be pumped again during the surface maneuver using more energy. Second, as Seaglider climbs it enters less dense water, becoming less positively buoyant and slowing down.

VBD pumping operations are allowed in the case of Seaglider being too heavy and slowing down. The **$MAX_BUOY** restriction does not apply to the climb phase. This usually does not affect the amount of energy used during the profile very much because the oil would need to be pumped during the surface maneuver anyway.

In the climb phase, Seaglider turns to starboard by banking to starboard and vice versa (as in aircraft flight).

When Seaglider reaches the depth **$D_SURF** it begins its approach to the surface. It computes how many more data samples to take, based on the observed vehicle vertical speed, depth and the data sample interval. The maximum number of data samples Seaglider may take from **$D_SURF** to the surface is 50. When the calculation is complete, Seaglider enters the passive G&C mode and collects the number of scientific data samples based on the above calculation. After this period of data acquisition, Seaglider enters the surface phase.

### Recovery

The recovery phase is entered either by command of the pilot (when it is necessary or desirable to keep Seaglider at the surface) or by an error condition detected by Seaglider's operating software. In the recovery phase, Seaglider stays on the surface and acquires a series of GPS fixes which are sent to the basestation so that Seaglider can be recovered.

In recovery, Seaglider enters a loop of obtaining a GPS fix and communicating that position with the basestation every **$T_RSLEEP** minutes. In practice, there are about two minutes of overhead in this process, so that the actual time between phone calls is closer to **$T_RSLEEP** + 2 minutes. This recovery loop may be exited by sending a **$RESUME** directive to Seaglider via the cmdfile. Once the **$RESUME** directive is received by Seaglider, it will start diving again.

CHAPTER 5

# *Piloting Parameters*

This chapter describes the command directives and parameters that govern the operation of the 1KA Seaglider. The following topics are covered:


- "Parameter Conventions" on page 83
- "Piloting Parameters" on page 85
- "Parameters by Category " on page 86


## *Parameter Conventions*

All parameters have a leading **$** in their name. Parameters are displayed in bold font, file names in italic. Nominal values are given with most parameters and are not default values. Parameter values reported by Seaglider in the log file associated with a dive include all the pilot-changeable parameters described in this document. In addition, the values generated on board Seaglider, such as glide angle, pitch angle and desired heading, are also given parameter-like names for consistency of parsing during post-dive data processing.

## Command File (cmdfile) State Directives

### *Introduction*

The command file (cmdfile) directives control the fundamental state of autonomous Seaglider operations. The diving state is the normal repeating sequence of canonical dives. The recovery state is when Seaglider is held at the surface, calling in to the basestation at pilot specified intervals with an updated position and receiving instructions.

Command file (cmdfile) directives are given as the last (and perhaps only) line of the command file and are stored on the basestation and transferred to Seaglider during its communication session. Directives do not have changeable values.

### *Directives*

#### $GO

**Definition:**

This command will cause Seaglider to continue in its current mode of operation. If in an autonomous run, doing repeated dives, it will continue to dive according to its current set of parameters.

If a **$GO** command is received while Seaglider is in the recovery state, Seaglider will stay in the recovery state. If received while Seaglider is in the diving state it will continue in that state. Note that error conditions that occur during a dive may cause the operating code to change the state of Seaglider from diving to recovery.

#### $RESUME

**Definition:**

This command will cause Seaglider to resume diving from within the recovery phase, using its current set of parameters.

If Seaglider is in dive state at the time the **$RESUME** command is received it will continue diving. If Seaglider is in recovery state at the time it receives a RESUME, it will start diving with existing parameters.

### $QUIT

**Definition:**

This command will cause Seaglider to go immediately to the recovery state.

Seaglider will hold at the surface, sleeping **$T_RSLEEP** minutes between the end of one communication session and the start of the next. There are about two minutes of communication overhead associated with each session, so the sessions are approximately (**$T_RSLEEP + 2**) minutes apart. It also fully inflates the bladder. This takes a couple minutes and consequently makes the time before the next transmission longer.

Replacing the **$QUIT** directive with a **$RESUME** directive will cause Seaglider to initiate a new dive with the existing set of parameters.

Table 5-1 outlines the effect of each directive on Seaglider in each of the two autonomous run states: diving and recovery.

**TABLE 5-1. Effect of Directive on Dive or Recovery State**

| Directive / State | $GO | $RESUME | $QUIT |
|---|---|---|---|
| Diving | Diving | Diving | Recovery |
| Recovery | Recovery | Diving | Recovery |

## *Piloting Parameters*

1KA Seaglider parameters are changeable by the pilot via the command file, using the **$name,value** (example: **$SM_CC,475**) convention. There is no space between the comma and the value. In this section, each parameter is listed alphabetically, described and where appropriate nominal, minimum and maximum values are given.

In the following section, “Most-Frequently Modified Parameters” on page 86, the parameters are listed in order of frequency used.

## *Parameters by Category*

For full details on the use of the parameters listed and how they interact with one another refer to “Alphabetized Parameters” on page 92.

### Most-Frequently Modified Parameters

#### *Dive Profile*

**$D_TGT**, (meters)

**$T_DIVE**, (minutes)

**$T_MISSION**, (minutes)

#### *Buoyancy Limits (use to save energy)*

**$MAX_BUOY**, (cc’s)

**$SM_CC**, (cc’s)

### Second Most-Frequently Modified Parameters

#### *Flight Behavior and Improvement Thereof*

**$C_VBD**, (AD counts)

**$C_PITCH**, (AD counts)

**$PITCH_GAIN**, (degrees/cm)

**$C_ROLL_DIVE**, (AD counts)

**$C_ROLL_CLIMB**, (AD counts)

**$D_SURF**, (meters)

**$GLIDE_SLOPE**, (degrees)

**$D_BOOST** (meters)

**$T_BOOST** (seconds)

#### *Communications*

**$T_RSLEEP**, (minutes)

**$CAPUPLOAD**, (Boolean)

#### *Navigation*

**$NAV_MODE**, (integer)

**$COURSE_BIAS**, (degrees)
**$HEAD_ERRBAND**, (degrees)
**$KALMAN_USE**, (integer)
**$FERRY_MAX**, (degrees)
**$HEADING**, (-1 or degrees)
**$SPEED_FACTOR**, (value - multiplier)
**$TGT_DEFAULT_LAT**, (degrees decimal minutes)
**$TGT_DEFAULT_LON**, (degrees decimal minutes)

### *Bottom Determination and Altimetry*

**$USE_BATHY**, (integer)
**$D_OFFGRID**, (meters)
**$ALTIM_BOTTOM_PING_RANGE**, (0 or meters)
**$ALTIM_BOTTOM_TURN_MARGIN**, (0 or meters)
**$ALTIM_PING_DEPTH**, (0 or meters)
**$ALTIM_PING_DELTA**, (0 or meters)

### *Dynamic Flight Feedback System*

**$PITCH_ADJ_GAIN**, (0/off or cm/deg)
**$PITCH_ADJ_DBAND**, (0/off or degrees)
**$ROLL_ADJ_GAIN**, (0/off or deg/seconds)
**$ROLL_ADJ_DBAND**, (0/off or degrees)

## Less Frequently-Modified Parameters

### *Flight Behavior and Improvement Thereof*

**$D_FLARE**, (meters)
**$APOGEE_PITCH**, (degrees)
**$ROLL_DEG**, (degrees)
**$D_FINISH**, (meters)
**$N_NOSURFACE**, (integer)
**$T_LOITER**, (seconds)

### *Turn Length and Data Sampling Rate during Turn*

**$T_TURN**, (seconds)

**$T_TURN_SAMPINT**, (seconds)

### *Communications and Files*

**$CALL_TRIES**, (integer)

**$CALL_WAIT**, (seconds)

**$N_FILEKB**, (integer)

**$CALL_NDIVES**, (integer)

**$UPLOAD_DIVES_MAX**, (integer)

**$COMM_SEQ**, (integer)

**$CAPMAXSIZE**, (bytes)

**$N_GPS**, (seconds)

**$T_GPS**, (minutes)

### *Bottom and Top Detection*

**$ALTIM_TOP_PING_RANGE**, (0 or meters)

**$ALTIM_TOP_TURN_MARGIN**, (0 or meters)

**$ALTIM_TOP_MIN_OBSTACLE**, (meters)

**$ALTIM_FREQUENCY**, (kHz)

**$ALTIM_PULSE**, (milliseconds)

**$ALTIM_SENSITIVITY**, (integer)

**$XPDR_VALID**, (integer)

**$XPDR_INHIBIT**, (seconds)

### *Flight Model and Environment*

**$HD_A**, (value)

**$HD_B**, (value)

**$HD_C**, (value)

**$RHO**, (gm/cc)

### *Safety*

**Caution:** Use caution when changing these parameters.

**$N_NOCOMM**, (integer)
**$D_ABORT**, (meters)
**$D_NO_BLEED**, (meters)
**$T_NO_W**, (seconds)
**$T_ABORT**, (minutes)
**$TGT_AUTO_DEFAULT**, (Boolean)
**$PITCH_MAXERRORS**, (integer)
**$ROLL_MAXERRORS**, (integer)
**$VBD_MAXERRORS,** (integer)
**$CF8_MAXERRORS**, (integer)
**$UNCOM_BLEED**, (integer)
**$RELAUNCH**, (integer)

### *Seaglider Modified*

**Caution:** Use caution when changing these parameters.

**$DIVE**, (integer)
**$MISSION**, (integer)
**$T_GPS_ALMANAC**, (minutes)
**$T_GPS_CHARGE**, (seconds)
**$R_PORT_OVSHOOT**, (AD counts)
**$R_STBD_OVSHOOT**, (AD counts)

## Parameters Set During Fabrication

### *Advanced Use Parameters*

**Caution:** Use caution when changing these parameters.

**$ID**, (integer)

**$MASS**, (grams)
**$T_WATCHDOG**, (minutes)
**$AH0_24V**, (AmpHours)
**$AH0_10V**, (AmpHours)
**$FILEMGR**, (integer)
**$PRESSURE_YINT**, (value)
**$PRESSURE_SLOPE**, (calibration value)
**$AD7714Ch0Gain**, (value)
**$TCM_PITCH_OFFSET**, (degrees)
**$TCM_ROLL_OFFSET**, (degrees)
**$ROLL_MIN**, (AD counts)
**$ROLL_MAX**, (AD counts)
**$ROLL_CNV**, (AD counts/degree)
**$ROLL_TIMEOUT**, (seconds)
**$ROLL_AD_RATE**, (AD counts/second)
**$PITCH_MIN**, (AD counts)
**$PITCH_MAX**, (AD counts)
**$PITCH_DBAND**, (cm)
**$PITCH_CNV**, (AD counts/cm)
**$P_OVSHOOT**, (cm)
**$PITCH_TIMEOUT**, (seconds)
**$PITCH_AD_RATE**, (AD counts/second)
**$VBD_MIN**, (AD counts)
**$VBD_MAX**, (AD counts)
**$VBD_DBAND**, (cc)
**$VBD_CNV**, (AD counts/cc)
**$VBD_TIMEOUT**, (seconds)
**$PITCH_VBD_SHIFT**, (cm travel / cc pumped)
**$VBD_PUMP_AD_RATE_SURFACE**, (AD counts/second)
**$VBD_PUMP_AD_RATE_APOGEE**, (AD counts/second)
**$VBD_BLEED_AD_RATE**, (AD counts/second)
**$HEAPDBG**, (Boolean)
**$INT_PRESSURE_SLOPE**, (calibration value)

**$INT_PRESSURE_YINT**, (calibration value)
**$DEVICE1**, (integer)
**$DEVICE2**, (integer)
**$DEVICE3**, (integer)
**$DEVICE4**, (integer)
**$DEVICE5**, (integer)
**$DEVICE6**, (integer)
**$COMPASS_USE**, (integer)
**$COMPASS_DEVICE**, (integer)
**$COMPASS2_DEVICE**, (integer)
**$PHONE_DEVICE**, (integer)
**$GPS_DEVICE**, (integer)
**$XPDR_DEVICE**, (integer)
**$SIM_W**, (off/0 or cm/seconds)
**$SIM_PITCH**, (off/0 or degrees)
**$SEABIRD_T_G**, (calibration value)
**$SEABIRD_T_H**, (calibration value)
**$SEABIRD_T_I**, (calibration value)
**$SEABIRD_T_J**, (calibration value)
**$SEABIRD_C_G**, (calibration value)
**$SEABIRD_C_H**, (calibration value)
**$SEABIRD_C_I**, (calibration value)
**$SEABIRD_C_J**, (calibration value)
**$STROBE** (integer)
**$MINV_10V** (voltage)
**$MINV_24V** (voltage)
**$LOGGERS** (integer)
**$LOGGERDEVICE1** (integer)
**$LOGGERDEVICE2** (integer)
**$LOGGERDEVICE3** (integer)
**$LOGGERDEVICE4** (integer)
**$CURRENT** (m/s, degrees, boolean)
**$MEM** (bytes)

**$FG_AHR_10V** (amp-hr)
**$FG_AHR_24V** (amp-hr)
**$FG_AHR_24Vo** (amp-hr)
**$FG_AHR_10Vo** (amp-hr)

### *Advanced GPCTD Parameters*

**Caution:** Use caution when changing these parameters.

**$PC_RECORDABOVE** (meters)
**$PC_PROFILE** (integer)
**$PC_XMITPROFILE** (integer)
**$PC_UPLOADMAX** (bytes)
**$PC_STARTS** (integer)
**$PC_INTERVAL** (seconds)

## Alphabetized Parameters

### $AD7714Ch0Gain (Set by manufacturer. Do not change.)

**Definition:**

The gain assigned to the pressure sensor channel on the AD7714 analog-to-digital converter. The parameter takes two values: 128 for normal Seaglider operations with the installed Paine pressure sensor, and 1 for bench testing where a synthetic voltage is injected in place of the pressure sensor output to simulate diving. If the parameters **$SIM_W** and **$SIM_PITCH** are non-zero, this parameter does not apply.

### $AH0_10V (Set by manufacturer. Do not change.)

**Definition:**

The capacity of the 10V (low voltage) battery pack (AmpHr). There is a small safety factor in this number, and its accuracy has been verified in post-recovery depletion testing of Seaglider battery packs. Seaglider goes into the recovery phase if the total 10V battery pack amp-hours used on a mission equals or exceeds this value.

**Nominal Value:** 95

**Minimum Value:** 1

**Maximum Value:** 100

### $AH0_24V (Set by manufacturer. Do not change.)

**Definition:**

The capacity of the 24V (high voltage) battery pack (AmpHr). There is a small safety factor in this number, and its accuracy has been verified in post-recovery depletion testing of Seaglider battery packs. Seaglider goes into the recovery phase if the total 24V battery pack amp-hours used on a mission equals or exceeds this value.

**Nominal Value:** 145

**Minimum Value:** 1

**Maximum Value:** 150

### $ALT_TEL_NUM

**Definition:**

The alternate telephone number Seaglider dials to connect to the basestation should it not be able to connect via the primary number, 13 digits maximum. This number is the PSTN number for the phone line connected to an alternate modem (if available) on a basestation for Seaglider operations. The format for the number is: international country code without leading zeros (for example, "1" for the US), then city/area code and number. There are no spaces or other interrupting characters between country code, city/area code or number.

The **$ALT...** mechanism allows for automatic switching between two telephone numbers in the event of a communication failure. If a communication session using the primary phone number (**$TEL_NUM**) does not successfully connect (after **$CALL_NDIVES** tries), the phone number is switched to the alternate number for the next surfacing.

If a communication session completes successfully on the alternate phone number, the phone number is switched back to the primary for the next surfacing.

**Note:** This parameter is not adjustable from the cmdfile. The number can be edited either through the pdoscmds.bat file (see *Extended PicoDOS Reference Manual,* writenv on page 279) or through direct connection to Seaglider via the serial communications cable (see "Checking the Primary and Alternate Phone Numbers" on page 52).

### $ALTIM_BOTTOM_PING_RANGE

**Definition:**

The range (in meters) from the presumed apogee depth (the nominal depth at which Seaglider begins its apogee maneuver) to ping for the bottom. Only one attempt is made to sound for the bottom unlike with **$ALTIM_PING_DEPTH**.

A value of 0 disables pinging.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1000

### $ALTIM_BOTTOM_TURN_MARGIN

**Definition:**

The distance (in meters) from the altimeter detected seafloor (or an obstacle) at which to initiate the apogee maneuver (bottom turn).

A value of 0 disables the use of the altimeter to determine the start of the apogee maneuver.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 100

### $ALTIM_FREQUENCY

**Definition:**

Frequency (kHz) to use for altimeter pings. The value must be an integer between 10 and 25.

**Nominal Value:** 13

**Minimum Value:** 10

**Maximum Value:** 25

### $ALTIM_PING_DELTA

**Definition:**

If the altimeter does not get a successful return and confirmation ping return at **$ALTIM_PING_DEPT**H, it continues to issue pings at depth intervals of **$ALTIM_PING_DELTA** meters.

See **$ALTIM_PING_DEPTH**.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1000

### $ALTIM_PING_DEPTH

**Definition:**

The depth of the first altimeter ping (meters), if non-zero.

If the altimeter gets a return, and a return to an immediate second confirmation ping, it sets the bottom depth equal to the current depth plus the altimeter range to the bottom.

The apogee maneuver is initiated at **$ALTIM_BOTTOM_TURN_MARGIN** meters above the bottom.

If **$ALTIM_PING_DEPTH** is non-zero, the altimeter timeout is set so that the maximum range is the larger of 0.75***$ALTIM_PING_DEPTH** and 1.2***$ALTIM_TOP_PING_RANGE**, if set. The first test is meant to exclude surface returns.

**Note: $ALTIM_PING_DEPTH** and the **$ALTIM_BOTTOM_PING_RANGE** modes are mutually exclusive.

If **$ALTIM_BOTTOM_PING_RANGE** is set, it is honored to the exclusion of **$ALTIM_PING_DEPTH**.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1000

### $ALTIM_PULSE

**Definition:**

Pulse width (ms) of altimeter pings. The value must be an integer between 1 and 9.

**Nominal Value:** 3

**Minimum Value:** 1

**Maximum Value:** 9

### $ALTIM_SENSITIVITY

**Definition:**

Sensitivity (volts) of the envelope detector on the altimeter. The value must be an integer between 0 and 5. A value of 0 disables the envelope detector, causing the altimeter to trigger on any return at the right frequency. Values between 1 and 5 require that the return signal be above that voltage for the duration of a pulse width before triggering.

**Nominal Value:** 2

**Minimum Value:** 0

**Maximum Value:** 5

### $ALTIM_TOP_MIN_OBSTACLE

**Definition:**

Minimum obstacle depth (in meters) to honor in initiating a subsurface finish.

**Nominal Value:** 2

**Minimum Value:** 0

**Maximum Value:** 100

### $ALTIM_TOP_PING_RANGE

**Definition:**

Range (in meters) from the surface (this is simply depth) at which to ping the altimeter. A value of 0 disables a ping.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 500

### $ALTIM_TOP_TURN_MARGIN

**Definition:**

The distance (in meters) from an altimeter detected obstacle at which to initiate the sub-surface finish. A value of 0 disables the use of the altimeter to determine the start of the sub-surface finish.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 100

### $APOGEE_PITCH

**Definition:**

Intermediate pitch (position of the pitch mass) that Seaglider pitches to between the dive and climb phases, to prevent stalling. Seaglider changes from whatever pitch angle it used for the dive phase to **$APOGEE_PITCH** when it observes a depth of greater than the apogee depth (whether by altimeter, **$D_TGT** or **$D_GRID**). The apogee maneuver also includes pumping the VBD to 0cc. After the apogee maneuver, the climb is initiated by changing both pitch and VBD to the opposite-signed values from the dive.

**Nominal Value:** -5

**Minimum Value:** -20

**Maximum Value:** 0

### $C_PITCH

**Definition:**

The center (neutral or flat) position (A/D counts) for pitch.

**Nominal Value:** 2700 (Actual initial value obtained from vehicle's trim sheet.)

**Minimum Value:** The software minimum of the system obtained from the vehicle's trim sheet.

**Maximum Value:** The software maximum of the system obtained from the vehicle's trim sheet.

### $C_ROLL_CLIMB

**Definition:**

The center (neutral or straight flight) position (A/D counts) for roll during the climb (positive pitch control) phase.

**Note:** The climb and dive roll centers will probably be different. This was implemented to correct for roll biases induced by physical asymmetries in Seaglider.

**Nominal Value:** 2025 (Actual initial value obtained from vehicle's trim sheet.)

**Minimum Value:** The software minimum of the system obtained from the vehicle's trim sheet.

**Maximum Value:** The software maximum of the system obtained from the vehicle's trim sheet.

### $C_ROLL_DIVE

**Definition:**

The center (neutral or straight flight) position (A/D counts) for roll during the dive (negative pitch control) phase.

**Note:** The climb and dive roll centers will probably be different. This was implemented to correct for roll biases induced by physical asymmetries in Seaglider.

**Nominal Value:** 2025 (Actual initial value obtained from vehicle's trim sheet.)

**Minimum Value:** The software minimum of the system obtained from the vehicle's trim sheet.

**Maximum Value:** The software maximum of the system obtained from the vehicle's trim sheet.

### $C_VBD

**Definition:**

The center (neutrally buoyant at a specified density) position (A/D counts) for VBD.

**Nominal Value:** 2900 (Actual initial value obtained from vehicle's trim sheet.)

**Minimum Value:** The software minimum of the system obtained from the vehicle's trim sheet.

**Maximum Value:** The software maximum of the system obtained from the vehicle's trim sheet.

### $CALL_NDIVES

**Definition:**

The number of profiles (dive/climb cycles) to perform before attempting communications. Seaglider normally surfaces following each dive, and GPS fixes 1 and 2 are obtained at each surfacing, independent of the value of **$CALL_NDIVES**.

**Nominal Value:** 1

**Minimum Value:** 1

**Maximum Value:** 10

### $CALL_TRIES

**Definition:**

The maximum number of phone calls to attempt during a surfacing between dives.

**Nominal Value:** 5

**Minimum Value:** 1

**Maximum Value:** 20

### $CALL_WAIT

**Definition:**

The wait time (seconds) between call attempts during a communication session. This is time to allow the Iridium satellite geometry to change and perhaps improve the connection.

**Nominal Value:** 60

**Minimum Value:** 0

**Maximum Value:** 600

### $CAPMAXSIZE

**Definition:**

Maximum size (in bytes, prior to compression) of the capture file to upload. If the capture file is to be uploaded (either due to critical output, **$CAPUPLOAD** set to 1, or a completed self test) and the capture file size exceeds this value, Seaglider creates a new capture file that is of the size requested, per the following strategy:

- If there are no critical lines of output, then the first **$CAPMAXSIZE** bytes are sent.
- If there are critical lines of output, then the new capture file consists of the first 20 of the critical output lines, with a window of output lines surrounding each critical line.

**Nominal Value:** 100000

**Minimum Value:** 1024

**Maximum Value:** 400000

### $CAPUPLOAD

**Definition:**

A Boolean value that determines if the capture file from the current dive should be uploaded or not. 0 means do not upload the capture file. 1 means upload the capture file. If critical output is captured during a dive, or a self test has been completed, then the capture file is uploaded regardless of the value of **$CAPUPLOAD**.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1

### $CF8_MAXERRORS (Set by manufacturer. Do not change.)

**Definition:**

The maximum number (integer) of Compact Flash (CF8) errors allowed before Seaglider goes into recovery phase. A CF8 error is counted against the **$CF8_MAXERRORS** limit when a CF8 open or write call continues to fail (returns an error code) after three retries.

**Nominal Value:** 20

**Minimum Value:** 0

**Maximum Value:** 500

### $COMM_SEQ

**Definition:**

The specification of the sequence of file transfer to use. A value of zero indicates the standard communication file transfer sequence: command (cmdfile), targets, science, current dive log file, current dive data file, earlier un-transferred log and data files, pdoscmds.bat, sgdddd.pz.nnn (the results of the pdoscmds.bat commands), and any other files as commanded in pdoscmds.bat.

A value of 1 indicates skipping the normal log and data file transmission and going directly to pdoscmds.bat, so the sequence for file transfers becomes command (cmdfile), targets, science, pdoscmds.bat, sgdddd.pz.nnn and any other files as commanded in pdoscmds.bat. This was implemented as a way of getting to the pdoscmds.bat file in the event that communications problems or file corruptions prevented data transfers. It is a control mode to be used only when communications or other Seaglider problems exist.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1

### $COMPASS_DEVICE (Set by manufacturer. Do not change.)

**Definition:**

Configuration flags (integer) specifying the specific model and port for the compass and transponder/altimeter devices. This integer value is equal to (port_number + 16*type_number). For example, for a TCM2-50 (type **0**) on general purpose port 1, **$COMPASS_DEVICE** = 1 + 16*0 = 1.

The array of available models is specific to each device. The compass device must be defined (parameter cannot be -1). For transponders not connected to a serial port (for example, Benthos ENT-380) the null port (generally index 0) can be specified. See **$DEVICE[1/2/3/4/5/6]**.

**Nominal Value:** 33

**Minimum Value:** 0

**Maximum Value:** 1023

### $COMPASS_USE (Set by manufacturer. Do not change.)

**Definition:**

An integer value that tells Seaglider which compass (**$COMPASS_DEVICE** or **$COMPASS2_DEVICE**) to use. (Not used at this time, because iRobot only puts one compass in Seaglider.)

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 2

### $COMPASS2_DEVICE (Set by manufacturer. Do not change.)

**Definition:**

Exactly the same as **$COMPASS_DEVICE**, except that it defines the second compass.

**Nominal Value:** 33

**Minimum Value:** 0

**Maximum Value:** 1023

### $COURSE_BIAS

**Definition:**

A heading bias (degrees) to compensate for a Seaglider's observed tendency to veer to one side. This value is subtracted from the desired heading to produce the target heading.

**Nominal Value:** 0

**Minimum Value:** -360

**Maximum Value:** 360

### $CURRENT (not user defined)

**Definition:**

The depth averaged current (m/s, degrees, Boolean validity check) calculated by the glider when using NAV_MODE, 2.

### $D_ABORT

**Definition:**

The maximum depth (meters) for Seaglider operations. If this depth is reached, the dive is aborted and Seaglider immediately enters the recovery phase.

**Nominal Value:** 1050

**Minimum Value:** 0

**Maximum Value:** 1100

### $D_BOOST

**Definition:**

The depth (meters) above which only the boost pump will run. If **$D_BOOST=0**, then both the boost pump and the main pump run simultaneously. If the value of **$D_BOOST** is greater than zero and Seaglider's depth is less than **$D_BOOST** when VBD begins pumping, only the boost pump will be used. If Seaglider VBD starts pumping at a depth greater than **$D_BOOST**, then the **$D_BOOST** parameter is ignored and both pumps are used. If the VBD engine is retried, both the boost and main pumps will be turned on during retry.

If a ***STANDARD BUOYANCY ENGINE*** is installed in the Seaglider, **$D_BOOST** must be set as follows:

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 120

Note: The boost pump on the standard buoyancy engine is not capable of pumping oil to increase buoyancy below 5m.

If an ***ENHANCED BUOYANCY ENGINE*** is installed in the Seaglider, **$D_BOOST** is used in conjunction with **$T_BOOST** and should be set as follows:

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 120

### $D_CALL

**Definition:**

A depth (meters) above which the glider will initiate the GPS acquisition and Iridium phone call portion of the surface maneuver. If this depth is not reached, a subsurface finish is executed. A value of 0 means the glider initiates the GPS acquisition and Iridium phone call at the surface.

**Nominal Value:** 0

### $D_FINISH

**Definition:**

The depth (meters) at which a dive is considered completed. Normally, this is 0, but can be a number greater than zero to specify the depth at which subsurface finish maneuvers should be started. Used only when an additional trigger to initiate a subsurface finish is present. (See **$N_NOSURFACE** on page 123.)

**Note:** If a subsurface finish has been triggered by **$N_NOSURFACE** and **$D_FINISH >= $D_SURF** then the dive will complete a subsurface finish. However, if a subsurface finish has been triggered by **$N_NO_SURFACE** and **$D_FINISH < $D_SURF** the dive will finish at the surface.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1000

### $D_FLARE

**Definition:**

The depth (meters) at which Seaglider flares to the computed pitch angle following the initial dive from the surface. The guidance and control (G&C) action at the start of the dive phase maintains full pitch forward as VBD bleeding takes place. If the desired VBD has already been reached, the bleed is skipped. A new G&C action is initiated as soon as a Seaglider reaches **$D_FLARE**: pitch is adjusted first (the flare), then VBD is adjusted (continued bleed to the target VBD), then roll is actuated to turn Seaglider to the correct heading.

**Nominal Value:** 3

**Minimum Value:** 0

**Maximum Value:** 990

### $D_GRID (not user defined)

**Definition:**

This parameter is calculated on board Seaglider during each dive based on the position and bathymetry data, but only if **$USE_BATHY** is set. If **$USE_BATHY** is -4, search for an on-board bathymap.nnn appropriate for the current position of Seaglider. This would be the standard usage in operating areas covered by more than one map. If **$USE_BATHY** is a positive integer then search for that particular on-board bathymap.

**Examples:**

- If **$USE_BATHY** = -4, then search all on-board bathymap.nnn files for one that covers the current Seaglider position.
- If **$USE_BATHY** = 7, then use the bathymetry file called bathymap.007 to get **$D_GRID**. No other map will be used.

### $D_NO_BLEED

**Definition:**

The depth (meters) below which Seaglider does not bleed (move) oil from the bladder into the internal reservoir on dives. This parameter is important during the deep dives because opening the bleed valve when there is a lot of pressure can cause it to get stuck in the open position.

**Nominal Value:** 200

**Minimum Value:** 1

**Maximum Value:** 600

### $D_OFFGRID

**Definition:**

The depth (meters) that the bathymetry map look-up routine returns in the event Seaglider's position is outside an area for which Seaglider carries a map.

**Nominal Value:** 100

**Minimum Value:** 10

**Maximum Value:** 1000

### $D_PITCH

**Definition:**

Depth (meters) that must be reached before the surface pitch maneuver is executed. If a depth shallower than this value is not reached then execute a subsurface maneuver. A value of 0 means the surface pitch maneuver is executed at the surface.

**Nominal Value:** 0

### $D_SAFE

**Definition:**

The target depth (meters) to use when flying an escape route and limiting the dive depth for VBD safety reasons. The escape is triggered by either VBD max errors exceeded or uncommanded bleed. If set to 0, the parameter is disabled.

**Nominal Value:** 0

### $D_SURF

**Definition:**

The depth (meters) at which Seaglider begins its approach to the surface. To collect data all the way to the surface, at **$D_SURF** Seaglider computes how many more data samples to take, based on the observed vehicle vertical speed, depth and the data sample interval. The number of additional points is limited to 50. Seaglider then goes into passive guidance and control (G&C) mode and collects that number of data points at the appropriate sample interval for the depth range. When complete, Seaglider enters the surface phase.

**Note:** This approach occasionally results in the last few data samples being taken when the conductivity sensor is actually in air, giving unrealistic conductivity values. These samples can be removed in shore side processing, if necessary.

**Nominal Value:** 2

**Minimum Value:** 0.5

**Maximum Value:** 10

### $D_TGT

**Definition:**

The nominal depth (meters) at which Seaglider begins the apogee phase, the transition from the negatively buoyant, pitch down dive to positively buoyant, pitch up climb. This depth is also used in conjunction with **$T_DIVE** to determine the specified vertical velocity for the dive and climb.

The actual depth of the starting point of the apogee maneuver can be determined by reading a digital bathymetric map (**$D_GRID**), the altimeter, or the pressure sensor. In either case, the vertical velocity specified by the combination of **$D_TGT** and **$T_DIVE** is retained by appropriate scaling of **$T_DIVE**. The apogee maneuver is not started until a depth greater than **$D_TGT** (or other depth trigger) is detected.

**Nominal Value:** 45

**Minimum Value:** 1

**Maximum Value:** 1000

### $DEEPGLIDER

**Definition:**

Indicates whether the glider is a standard Seaglider or a Deepglider. A zero indicates the glider is standard; a 1 indicates the glider is a deep glider.

**Note:** All Seagliders produced by iRobot Corporation are standard gliders with a maximum depth capability of 1000m.

### $DEEPGLIDERMB

**Definition:**

A Boolean value that indicates whether the main board is intended for use in Deepglider. A zero means the motherboard is intended for a standard glider while a 1 means the motherboard is intended for a deep glider.

**Note:** All Seagliders produced by iRobot Corporation are standard gliders with a maximum depth capability of 1000m.

### $DEVICE[1/2/3/4/5/6] (Set by manufacturer. Do not change.)

**Definition:**

Configuration flags specifying device type and port for each of the six possible attached science sensors. Empty device slots are indicated with a parameter value of -1. Non-negative integer entries indicate that a device is attached. The encoding is specific to the version of Seaglider's software. These entries are set through Seaglider's menu system by the builders/assemblers.

**Minimum Value:** -1

**Maximum Value:** 104

### $DIRECT_CONTROL

**Definition:**

Allows direct control of the pitch and buoyancy settings of the glider

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1

### $DIVE

**Definition:**

The number of the next dive.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 9999

### $EBE_ENABLE

**Definition:**

Enables Enhanced Buoyancy Engine (EBE) functionality. A value of 1 turns this functionality on.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 1

### $ESCAPE_HEADING

**Definition:**

The base heading the Seaglider will steer in an escape recovery situation when either no position fix is available or no escape target was supplied in the targets file.

### $ESCAPE_HEADING_DELTA

**Definition:**

The actual heading steered by the Seaglider in an escape recovery situation will always be $ESCAPE_HEADING +/- $ESCAPE_HEADING_DELTA. The sign will switch (and thus the heading will toggle) when the bottom depth (as detected by altimetry or $T_NO_W) shallows by 5% relative to the depth at the last toggle.

### $FERRY_MAX

**Definition:**

Maximum correction (degrees) to apply to the rhumb line to the active (next) waypoint when **$NAV_MODE = 2**. This is a safety limit to prevent spurious depth-averaged current calculations from giving Seaglider a heading in the wrong direction.

**Nominal Value:** 45

**Minimum Value:** 0

**Maximum Value:** 90

### $FILEMGR (Set by manufacturer. Do not change.)

**Definition:**

An integer parameter that specifies how aggressively to manage the onboard file system. 0 = no file management; 1 = only store compressed files; 2 = delete splits on failed phone call.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 2

### $FIX_MISSING_TIMEOUT (Set by manufacturer. Do not change.)

**Definition:**

An integer used only for debugging.

### $GLIDE_SLOPE

**Definition:**

The absolute value of the maximum glide slope (degrees) allowed for Seaglider. The glide slope is calculated on board Seaglider to best achieve the goals of the next dive. The stall angle provides the lower limit; this parameter is the upper limit.

**Nominal Value:** 30

**Minimum Value:** 10

**Maximum Value:** 90

### $GPS_DEVICE (Set by manufacturer. Do not change.)

**Definition:**

A configuration value specifying the model of the attached GPS device. These devices have dedicated hardware ports on all motherboard revisions and as such a port specification is not necessary. A GPS device must be defined (cannot be -1).

**Nominal Value:** 32

**Minimum Value:** 0

**Maximum Value:** 1023

### $HD_A

**Definition:**

The hydrodynamic parameter representing the lift coefficient, determined empirically and used in Seaglider's on-board performance prediction and guidance calculations.

**Nominal Value:** 0.003836

**Minimum Value:** 0.001

**Maximum Value:** 0.005

### $HD_B

**Definition:**

The hydrodynamic parameter representing the drag coefficient, determined empirically and used in Seaglider's on-board performance prediction and guidance calculations.

**Nominal Value:** 0.010078

**Minimum Value:** 0.001

**Maximum Value:** 0.02

### $HD_C

**Definition:**

The hydrodynamic parameter representing the induced drag coefficient, determined empirically and used in Seaglider's on-board performance prediction and guidance calculations.

**Nominal Value:** 0.00000985

**Minimum Value:** 0

**Maximum Value:** 0.0001

### $HEAD_ERRBAND

**Definition:**

Deadband for heading (degrees). This value is used to determine if a correction to heading is required during an active guidance and control (G&C) mode. If the absolute value of the difference between the actual heading and the desired heading is less than or equal to **$HEAD_ERRBAND**, no heading correction is made. If the difference is greater than **$HEAD_ERRBAND**, then a turn is performed until the heading is passed, or until the amount of time **$T_TURN** has elapsed.

**Nominal Value:** 10

**Minimum Value:** 0

**Maximum Value:** 180

### $HEADING

**Definition:**

Floating point value between -1.0 and 360.0 (true degrees, 0.0 and 360.0 are equivalent values).

Used in conjunction with certain navigation modes (controlled by the **$NAV_MODE** parameter) to determine the course steered by the Seaglider. If **$NAV_MODE** is 0, 1, or 2 and the value of **$HEADING** is between 0.0 and 360.0, the glider will use this value to synthesize a waypoint 20 km distant on the specified bearing from the current location. If **$NAV_MODE** is 3, **$HEADING** is added to the depth-averaged current calculated for the previous dive, to give a Seaglider heading that is the specified amount to the right of the current.

**Note:** If heading is not being used by Seaglider to navigate make sure that **$HEADING,-1** is used.

**Nominal Value:** -1

**Minimum Value:** -1

**Maximum Value:** 360

### $HEAPDBG (Set by manufacturer. Do not change.)

**Definition:**

A Boolean value set during fabrication or building that is only used for debugging.

### $ID (Set by manufacturer. Do not change.)

**Definition:**

Seaglider identification (serial) number. This is an integer between 1 and 999. Leading zeros are not required. This identification number is used in many ways, including creating Seaglider's login on the basestation, in file naming conventions and as a serial number for manufacturing purposes.

**Nominal Value:** Set at factory (>500).

**Minimum Value:** 1

**Maximum Value:** 999

### $INT_PRESSURE_SLOPE (Set by manufacturer. Do not change.)

**Definition:**

The slope (psia per A/D count) calibration of the internal pressure sensor. The sensor has a span of 0 to 30 psia, with a 90mV output at full-scale at 12V excitation. The output is proportional to the supply. Seaglider's excitation is 4.096V and the gain is 100, so 30 psia = 90*4.096/12*100*1. Full-scale = 90*4.096*gain*counts/mV, so the nominal slope is 0.009766 psia per A/D count.

**Nominal Value:** 0.009766

**Minimum Value:** 0.001

**Maximum Value:** 1

### $INT_PRESSURE_YINT (Set by manufacturer. Do not change.)

**Definition:**

The y-intercept of the linear calibration of the internal pressure sensor.

**Nominal Value:** 0

**Minimum Value:** -5

**Maximum Value:** 5

### $KALMAN_USE

**Definition:**

The control parameter for the run state of the Kalman filter navigation program. The **$NAV_MODE** parameter controls whether the Kalman filter output heading is used to control Seaglider. This separation of functions allows the Kalman filter to be run, but not used, while it "learns" the currents.

0 Reset the Kalman state vector and origin of local Kalman coordinate system to 0 and restart the filter.

1 Run the Kalman filter.

2 Do not run the Kalman filter.

**Examples:**

- If **$KALMAN_USE** is 0 or 1, and **$NAV_MODE** is 1, the **$KALMAN_USE** filter results are used to determine Seaglider's heading.
- If **$NAV_MODE** is 1, but **$KALMAN_USE is 2** (not being run), Seaglider acts as though **$NAV_MODE** is 0.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 2

### $LENGTH

**Definition:**

Length in meters of the Seaglider direct control. Not to be changed by the user.

### $LOGGERS

**Definition:**

A bit mask that tells Seaglider which of the autonomous logging devices configured with the Autonomous Logger Interface (ALI) to use. When **$LOGGERS** is set to 0, no ALI devices will be run during self tests or dives. When **$LOGGERS** is set to >0, the installed ALI sensors are turned on according to the bitmask below. An 'X' means the ALI sensor is running.

| $LOGGER value | Sensor 1 | Sensor 2 |
|---|---|---|
| 0 | | |
| 1 | X | |
| 2 | | X |
| 3 | X | X |

### $MASS (Set by manufacturer. Do not change.)

**Definition:**

The mass of Seaglider in grams. This value is used in on-board buoyancy and current estimation calculations.

**Nominal Value:** 52000

**Minimum Value:** 50000

**Maximum Value:** 54000

### $MAX_BUOY

**Definition:**

The absolute value of the maximum negative thrust (in cc) that Seaglider is allowed to develop during the dive phase. There is no restriction on buoyancy during the climb phase.

**Nominal Value:** 150

**Minimum Value:** 0

**Maximum Value:** 600

### $MEM

**Definition:**

An output from the glider specifying the amount of free memory in bytes.

### $MINV_10

**Definition:**

The minimum allowable observed voltage on the 10 volt battery pack before the glider will stop diving and go into recovery. A zero disables the check.

**Nominal Value:** 8

### $MINV_24

**Definition:**

The maximum allowable observed voltage on the 24 volt battery pack before the glider will stop diving and go into recovery. A zero disables the check.

**Nominal Value:** 19

### $MISSION

**Definition:**

The current Seaglider mission number. This value is intended to be unique for each of a particular Seaglider's deployments. It has no effect on Seaglider operations, but is reported back in data files for the purpose of data file distinction from other missions a Seaglider may have performed.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 999

### $MOTHERBOARD (Set by manufacturer. Do not change.)

**Definition:**

An integer value indicating the motherboard revision carried by Seaglider. This value is set at the factory by the builder or assembler.

### $N_FILEKB

**Definition:**

An integer value (**$N_FILEKB**), which is the size (in kilobytes) and type (gzip-compressed or uncompressed) of file used for data uploading. Positive numbers direct Seaglider to first use gzip to compress the data file, then split it into **$N_FILEKB** -sized pieces. Negative values for **$N_FILEKB** disable the gzip compression, but still cause the non-gzipped binary data file to be split into **$N_FILEKB**-sized pieces before transmission. A value of 0 means no splitting or compression is performed. The maximum allowed value of this parameter is currently set to 16.

**Nominal Value:** 4

**Minimum Value:** -16

**Maximum Value:** 16

### $N_GPS

**Definition:**

The maximum number of seconds to wait for a GPS fix with HDOP = 2.0 or less. If no such fix is acquired, the last (most recent) GPS fix is used, which is probably, but not necessarily, the most accurate fix available in the specified time period.

**Nominal Value:** 20

**Minimum Value:** 1

**Maximum Value:** 60

### $N_NOCOMM

**Definition:**

The number of dives that are allowed to occur without a complete and successful data communication session, before the surface buoyancy parameter **$SM_CC** is set to the maximum allowed by the software limits. This is a safety provision in the event **$SM_CC** is not sufficient to allow for a good antenna position. Also, after **$N_NOCOMM** is reached, Seaglider will use the alternate phone number to call the basestation.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 10

### $N_NOSURFACE

**Definition:**

An integer value that determines when the Seaglider will finish the dive at **$D_FINISH** and when it will finish at the actual surface. For values greater than 1, the rule is that when the remainder of **$DIVE** divided by **$N_NOSURFACE** is zero, Seaglider will finish the dive at depth **$D_FINISH**. Other dives finish at the surface. For negative values less than -1, this logic is reversed and the values of **$DIVE** divisible by the absolute value of **$N_NOSURFACE** finish at the surface; all others will be subsurface finishes.

In addition to **$N_NOSURFACE** not having a remainder for the maneuver to be initiated, **$D_FINISH** must be >= **$D_SURF** for Seaglider to complete a subsurface finish. If a subsurface finish has been triggered by **$N_NOSURFACE** and **$D_FINISH** < **$D_SURF** the dive will end at the surface.

Values of 1 and -1 are not allowed. A value of 0 disables this behavior.

**Note:** Other considerations when using $N_NOSURFACE; When **$N_NOSURFACE** is not equal to 0, Seaglider will not acquire GPS fixes when it completes a dive subsurface.

As a result, **$NAV_MODE** 1, 2 and 3 should not be used when **$N_NOSURFACE** is not equal to zero as they will not produced the desired results. Only **$NAV_MODE 0**, which causes the glider to steer a specific heading, should be used when **$N_NOSURFACE** is not equal to zero.

Additionally, when **$NAV_MODE** is set to zero, the **$HEADING** parameter must be set to a value between 0 and 360.

For example, if the **$N_NO_SURFACE** is set to 5, **$NAV_MODE** is set to 0 and **$HEADING** is set to 90, then Seaglider will surface and acquire a GPS position and initiate a communication session with the basestation only on dives that are evenly divisible by 5. On all dives, the glider will point its nose due east and will not be able to compensate for currents.

If GPS positions need to be obtained on each dive to enable all **$NAV_MODE** options yet surface time needs to be minimized, **$CALL_NDIVES** can be set to a value greater than 1. This directs the glider to surface and obtain a GPS fix after each dive but not attempt to establish a communication session with the basestation until the number of dives since the last communication equals **$CALL_NDIVES**.

**Nominal Value:** 0

**Minimum value:** -10

**Maximum Value:** 10

### $NAV_MODE

**Definition:**

An integer value specifying the method used to choose a heading for Seaglider to maintain on the next dive, according to the following:

0 Steer constant heading **($HEADING)**

1 Kalman filter (see **$KALMAN_USE**), Seaglider uses the past current information gathered by the Kalman filter to correct Seaglider's flight path for the next dive.

2 Ferry angle correction with respect to the calculated depth-averaged current.

3 Steer relative to the depth-averaged current.

**Note:** If heading is set to something other than -1, it WILL be used instead of waypoint in ANY mode. If the pilot wants to fly by waypoint rather than heading, **$HEADING** must be set to -1.

Navigation flow proceeds as follows:

1. If a valid depth-averaged-current (DAC) has been calculated and **NAV_MODE** = 3 (steer relative to DAC), synthesize a heading = current direction + **$HEADING**.
2. If a synthesized heading exists from (1) or **$HEADING** >= 0, synthesize a target 20 km away at a true bearing given by that heading. Otherwise, choose a target based on the usual selection process. Calculate range and bearing to the target.
3. Modify the dive time and depth as needed to honor safeties and grids. Calculate speed limits - Seaglider minimum and maximum possible horizontal speeds through the water. The speed limits and dive times fix the range of possibilities for Seaglider's distance through the water.
4. Choose a glide slope that gets Seaglider closest to the target (maximum speed if the target is distant, minimum speed if the target is too close).

5. At this point Seaglider has a bearing and glide slope in hand. If **$NAV_MODE** = 0 or **$NAV_MODE** = 3, then these are the heading and slope for this dive.
6. If **$NAV_MODE** = 2 (calculate set/ferry angle correction) and we have calculated a valid DAC, we apply **$SPEED_FACTOR** to the speed limits to account for the fact that Seaglider actually won't achieve its ideal speed over the entire dive. We then iteratively calculate set corrections as a function of our horizontal speed through the water so that we optimize Seaglider's travel toward the target (just as we chose our slope above). At each speed setting, the code uses a nonlinear solver to solve for ferry angle. First guess at speed is maximum. If the predicted distance over ground (DOG) with the set correction is less than the range to target (typical case) then the computation is complete. Otherwise we try the minimum speed. If the resulting DOG is greater than the range to target then the computation can't do any better and it is complete. If neither limit applies, we iterate via bisection to settle on the best speed. At convergence, Seaglider has a ferry angle to steer and a horizontal speed to apply. The predicted horizontal speed is used to calculate a new value for glide slope.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 3

### $P_OVSHOOT (Set by manufacturer. Do not change.)

**Definition:**

The distance (cm) by which the pitch mass is allowed to overshoot its target after the pitch motor is turned off.

**Nominal Value:** 0.04

**Minimum Value:** -0.1

**Maximum Value:** 0.1

### $PC_RECORDABOVE

**Definition:**

A pumped CTD command that sets the depth (in meters) above which the sensor will turn sample. A value of 0 turns the sensor off.

**Minimum Value:** 0

**Maximum Value:** 1000

### $PC_PROFILE

**Definition:**

A pumped CTD command that specifies when the CTD will record data: none, downcast only, upcast only or both down- and upcast.
0 - none
1 - downcast only
2 - upcast only
3 - down- and upcast

### $PC_XMITPROFILE

**Definition:**

A pumped CTD command that specifies which data profiles from a dive are transmitted to the basestation: none, downcast only, upcast only or both down- and upcast.
0 - none
1 - downcast only
2 - upcast only
3 - down- and upcast

### $PC_UPLOADMAX

**Definition:**

The **$PC_UPLOADMAX** parameter is currently unused by the GPCTD

**Minimum Value:** 0, no data uploaded

### $PC_STARTS

**Definition:**

A diagnostic value output by the pumped CTD, that keeps track of the number of times the sensor restarts during a mission. There should be 2 restarts per dive, one for the downcast and one for the upcast.

### $PC_INTERVAL

**Definition:**

A pumped CTD command that specifies the sampling interval in seconds. 1-4 second sampling intervals: The CTD is in Continuous Sampling Mode. The pump and all sampling circuitry remain on continuously. Power consumption for any of these sampling intervals is the same. However, memory usage decreases with increasing sampling interval.

5-14 second sampling intervals: The CTD is in Fast Sampling Mode. The pump runs continuously and measurements are made at the chosen interval.

15-3600 second sampling intervals: The CTD is in Slow Interval Sampling Mode. In this mode CTD samples are taken but DO samples are not. The pump runs for 11.3 seconds prior to a measurement and an additional 2.1 seconds during the measurement. In-between sampling intervals the pump is off and the CTD is in low power state.

**Nominal Value:** 1

### $PHONE_DEVICE (Set by manufacturer. Do not change.)

**Definition:**

A configuration value specifying the model of the attached device, set by the assembler or builder. These devices have dedicated hardware ports on all motherboard revisions and, as such, a port specification is not necessary.

**Nominal Value:** 48

**Minimum Value:** 0

**Maximum Value:** 1023

### $PITCH_AD_RATE (Set by manufacturer. Do not change.)

**Definition:**

The pitch rate (A/D counts/second) used as the threshold for retries when pitching. If the observed rate is less than this number, the pitch motor is stopped and restarted. The retries continue until the pitch motor timeout limit is reached, then an error is declared.

**Nominal Value:** 175

**Minimum Value:** 0

**Maximum Value:** 200

### $PITCH_ADJ_DBAND

**Definition:**

This parameter (degrees), with **$PITCH_ADJ_GAIN**, enables and adjusts active (closed-loop) control on Seaglider pitch during a dive and climb. Seaglider automatically seeks to maintain the pitch angle by moving the pitch mass when:

| Pitch observed - Pitch desired | **>$PITCH_ADJ_DBAND**

**Note:** A value of 0 disables automatic pitch adjustment.

**Nominal Value:** 1 (if in use)

**Minimum Value:** 0

**Maximum Value:** 40

### $PITCH_ADJ_GAIN

**Definition:**

This parameter, with **$PITCH_ADJ_DBAND**, enables and adjusts active (closed-loop) control on Seaglider pitch during a dive and climb. The amount of the adjustment is given by:

(Pitch Desired - Pitch Observed)***$PITCH_ADJ_GAIN**.

**$PITCH_ADJ_GAIN** has units of cm/degree. Adjustments are calculated at the beginning of the active guidance and control (G&C) phase, based on the pitch observed over the same samples for which observed vertical speed is calculated. Adjustments are not made during the first two active G&C phases following the start of a dive or climb. A value of zero disables automatic pitch adjustment.

**Nominal Value:** 0.045 (if in use)

**Minimum Value:** 0

**Maximum Value:** 1000

### $PITCH_CNV (Set by manufacturer. Do not change.)

**Definition:**

The pitch position conversion factor, from A/D counts to centimeters (cm/AD count). This is a constant determined by the pitch of the worm gear that drives the pitch motion, and is set at the factory by the builder or assembler.

**Value:** 0.003125763

### $PITCH_DBAND

**Definition:**

The pitch position deadband (cm) within which no further pitch motion will be commanded.

**Nominal Value:** 0.05

**Minimum Value:** 0

**Maximum Value:** 1

### $PITCH_GAIN

**Definition:**

The amount of vehicle pitch (degrees) change corresponding to a 1 cm movement of the pitch mass.

**Nominal Value:** 30

**Minimum Value:** 15

**Maximum Value:** 40

### $PITCH_MAX (Set by manufacturer. Do not change.)

**Definition:**

Pitch position software limit (A/D counts) aft.

**Typical Value:**4000 (Value determined by the software maximum of the system. Obtain actual value from the vehicle's trim sheet).

### $PITCH_MAXERRORS (Set by manufacturer. Do not change.)

**Definition:**

The number of pitch motor errors allowed before Seaglider goes into recovery phase. An error occurs when the **$PITCH_TIMEOUT** expires prior to achieving the commanded pitch A/D position.

**Caution:** An error in the pitch could lead to loss of Seaglider!

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 100

### $PITCH_MIN

**Definition:**

Pitch position software limit (A/D counts) forward. This is also the value to which the pitch mass moves during the surface maneuver (fully forward for maximum pitch down).

**Typical Value:**100 (Value determined by the software minimum of the system. Obtain actual value from the vehicle's trim sheet.)

### $PITCH_TIMEOUT (Set by manufacturer. Do not change.)

**Definition:**

Pitch mass timeout (seconds).

**Nominal Value:** 16

**Minimum Value:** 15

**Maximum Value:** 25

### $PITCH_VBD_SHIFT (Set by manufacturer. Do not change.)

**Definition:**

The pitch compensation ($cm/cm^3$) required to balance the mass of the hydraulic oil moving forward and aft with the change in buoyancy as a result of VBD changes.

**Nominal Value:** 0.00123

**Minimum Value:** 0

**Maximum Value:** 0.01

### $PRESSURE_SLOPE (Set by manufacturer. Do not change.)

**Definition:**

Slope of linear fit between psig and pressure sensor output (after digitization to A/D counts through AD7714). The fit is calculated from calibration data received with each pressure sensor, and converted to A/D counts knowing the configuration of the AD7714 and associated circuitry. This number is a constant for each pressure sensor and associated calibration.

**Typical Value:** $1.16 \times 10^{-4}$

### $PRESSURE_YINT

**Definition:**

Y-intercept of linear fit between psig and pressure sensor output (after digitization to A/D counts through AD7714). This is the value that is adjusted in the field at launch to correct the pressure sensor relative to atmospheric pressure so that the seawater surface corresponds to 0m depth.

**Typical Value:** -19.65

**Minimum Value:** -50

**Maximum Value:** 50

### $R_PORT_OVSHOOT

**Definition:**

Roll mass overshoot (A/D counts) to port after roll motor is turned off. Overshoots are assumed to be positive, in the sense of past the desired position. The sign of the **$R_PORT_OVSHOOT** parameter indicates how the code handles the overshoots, not a direction.

Positive values allow Seaglider's operating software to automatically compute the roll overshoots and apply them after each roll maneuver. Negative values allow the pilot to specify a static overshoot value to be applied uniformly to each roll maneuver. In the case where Seaglider is computing and applying the roll overshoots, the value reported in the log file is the last value computed during a dive.

**Nominal Value:** 25

**Minimum Value:** -100

**Maximum Value:** 100

### $R_STBD_OVSHOOT

**Definition:**

Roll mass overshoot (A/D counts) to starboard after motor is turned off. See **$R_PORT_OVSHOOT** for more detail.

**Nominal Value:** 25

**Minimum Value:** -100

**Maximum Value:** 100

**$RELAUNCH** (**Do not change!!** Default value of 0 set by manufacturer. Loss of glider could result if this parameter is changed.)

**Definition:**

The **$RELAUNCH** parameter controls the behavior of the Seaglider when an reboot condition occurs. When **$RELAUNCH** is 1 and a reboot occurs the Seaglider will continue its current missions. When **$RELAUNCH** is 0 and a reboot occurs the Seaglider will enter Recovery.

**$RHO**

**Definition:**

The water density (kg/L) used for converting buoyancy force in grams to seawater displacement in $cm^3$. This parameter is also used in the on-board performance prediction computations.

**Nominal Value:** 1.0275

**Minimum Value:** 1

**Maximum Value:** 1.04

**$ROLL_AD_RATE**

**Definition:**

The roll rate in A/D counts per second that Seaglider's operating code uses as the threshold for retries when rolling. If the observed rate is less than this number, the roll is stopped and restarted. The retries continue until the roll motor timeout limit is reached, then an error is declared.

**Nominal Value:** 350

**Minimum Value:** 100

**Maximum Value:** 450

## $ROLL_ADJ_DBAND

**Definition:**

This parameter, in conjunction with **$ROLL_ADJ_GAIN**, controls the automatic adjustment of Seaglider's roll centers based on observed turn rate. At the end of a complete passive phase, a full guidance and control (G&C) interval, Seaglider adjusts the appropriate dive or climb roll center based on the turn rate over the last half of the passive phase if:

|turn rate| > **$ROLL_ADJ_DBAND**

**Note: $ROLL_ADJ_DBAND** has units of degrees/second. A value of zero disables automatic adjustment of the roll centers.

**Nominal Value:** 0.03

**Minimum Value:** 0

**Maximum Value:** 1000

### $ROLL_ADJ_GAIN

**Definition:**

This parameter, with **$ROLL_ADJ_DBAND**, controls the automatic adjustment of Seaglider's roll centers based on observed turn rate. The amount of the adjustment is:

-(turn rate)***$ROLL_ADJ_GAIN**, if climbing, and

(turn rate)***$ROLL_ADJ_GAIN**, if diving.

**Note:** Turn rate has units of degrees/second. This adjustment is only meaningful the next time Seaglider turns, as Seaglider rolls back to this new neutral position on the next turn. It will not immediately roll from the old neutral position to the new neutral position at the start of the next active guidance and control (G&C) phase.

**Note:** If **$HEAD_ERRBAND** is large, then several active/passive G&C phases might pass before a turn is initiated and the new roll center is used. A new adjustment is computed only after a turn has been completed.

**Nominal Value:** 1.0

**Minimum Value:** 0

**Maximum Value:** 1000

### $ROLL_CNV (Set by manufacturer. Do not change.)

**Definition:**

Roll position conversion factor, from A/D counts to degrees. This is a constant determined by the design of the roll gear train, motor and potentiometer.

**Value:** 0.02827

### $ROLL_DEG

**Definition:**

The number of degrees to roll the internal roll mass during a turn.

**Nominal Value:** 40

**Minimum Value:** 0

**Maximum Value:** 60

### $ROLL_GAIN_P

**Definition:**

Allows proportional coefficient for closed-loop heading control

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 5

### $ROLL_MAX (Set by manufacturer. Do not change.)

**Definition:**

Roll position software limit (A/D counts) to starboard. Seaglider's operating software stops the mass shifter at this value when rolling to starboard.

**Typical Value:**4000 (Value determined by the software maximum of the system. Obtain actual value from the vehicle's trim sheet.)

### $ROLL_MAXERRORS (Set by manufacturer. Do not change.)

**Definition:**

The number of roll motor errors allowed before Seaglider goes into recovery phase. An error occurs when the **$ROLL_TIMEOUT** expires prior to achieving the commanded roll A/D position.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 100

### $ROLL_MIN (Set by manufacturer. Do not change.)

**Definition:**

Roll position software limit (A/D counts) to port. Seaglider's operating software stops the mass shifter at this value when rolling to port.

**Typical Value:**50 (Value determined by the software minimum of the system. Obtain actual value from the vehicle's trim sheet.)

### $ROLL_TIMEOUT (Set by manufacturer. Do not change.)

**Definition:** Roll maneuver timeout (seconds).

**Nominal Value:** 15

**Minimum Value:** 10

**Maximum Value:** 20

### $SEABIRD_[C_G/ C_H/ C_I/ C_J/ T_G/ T_H/ T_I/ T_J] (Set by manufacturer. Do not change.)

**Definition:**

Sea-Bird Electronics provided calibration coefficients for their conductivity and temperature sensor on Seaglider. These values are used to compute calibrated temperature and salinity for hardware test purposes and in situ density for self-trimming applications. Currently used for subsurface finish maneuvers in which Seaglider attempts to become neutral at a fixed depth below the surface.

Parameters that begin with **$SEABIRD** may be flagged by Seaglider during self test as being out of range. The acceptable parameter ranges have changed since ranges used for the check were written into the software. For the Sea-Bird parameters ONLY this warning can be ignored. The parameters are installed at the factory, based on the calibration sheets delivered with the CT sensor and should not be changed.

### $SIM_PITCH

**Definition:**

Simulated Seaglider pitch angle (degrees) during the dive phase of a simulated run. If non-zero, this value is used in place of observed pitch on the dive phase of a simulated run. For simulated dives, a **$SIM_PITCH** value of -20 is often used. A value of 0 disables this feature. This parameter is automatically zeroed during the Sea Launch procedure.

### $SIM_W

**Definition:**

Simulated Seaglider vertical velocity (m/s). If non-zero, this value is used to generate depths so that Seaglider can do simulated dives in the lab or on the deck. For simulated dives, a **$SIM_W** value of 0.1 is often used. A value of 0 disables this feature. This parameter is automatically zeroed during the Sea Launch procedure.

### $SM_CC

**Definition:**

The specified minimum buoyancy value of the VBD ($cm^3$) that Seaglider attains at the surface. If Seaglider enters the surface maneuver with less than the minimum buoyancy value of VBD, it pumps to this value. However, if Seaglider enters the surface maneuver with more than the minimum buoyancy value specified, it does not change the value of VBD and continues to the next part of the surface maneuver.

**Typical Value:**250

**Minimum Value:** 150

**Maximum Value:** 700

### $SPEED_FACTOR

**Definition:**

A factor to compensate for Seaglider's inability to dive at the desired horizontal velocity. This is a measure of the efficiency of Seaglider's progress along a specified track. Factors that lower Seaglider's efficiency are turns, leaving the surface at arbitrary headings and reduced horizontal speed during the apogee maneuver. **$SPEED_LIMITS** are multiplied by this factor and **$KALMAN_CONTROL** components are divided by it.

**Nominal Value:** 1

**Minimum Value:** 0.1

**Maximum Value:** 1

### $T_ABORT

**Definition:**

A safety time (minutes) such that if the elapsed time on a given dive exceeds this time, Seaglider enters the recovery state.

**Nominal Value:** 1440

**Minimum Value:** 720

**Maximum Value:** 4320

### $T_BOOST

**Definition:**

Time (seconds) to run the boost pump in. The boost pump can either operate by itself in water depths less than 120 m or prime the main pump for pumping operations in the complete glider operating depth range. This parameter is used in conjunction with **$D_BOOST**. **$T_BOOST** must be set to zero if an EBE is not installed in the glider.

$T_BOOST = 0, then the boost pump is used for the entire pumping operation. If $T_BOOST is >0, the boost pump runs by itself for the first 2 seconds. Then, both pumps run simultaneously for the remainder of **$T_BOOST** seconds. At the end of **$T_BOOST** seconds, the boost pump turns off while the main pump continues to run.

**Note:** It is recommended that if the desired **$T_BOOST** is >0 that the value be at least three seconds. This allows a one second operational overlap of the two pumps before the boost pump turns off.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 20

### $T_BOOST_BLACKOUTS

**Definition:**

Time in seconds to wait after boost pump is disabled before allowing a retry. Not to be changed by the user.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 30

### $T_DIVE

**Definition:**

The time (minutes) for Seaglider to make one dive-climb cycle to the depth **$D_TGT** and back to the surface. This value does not include the time for pumping during the apogee phase. The value is used to calculate Seaglider's desired vertical velocity in a particular dive, using the naive calculation:

$w_d$ (cm/s) =2***$D_TGT***100/(***$T_DIVE***60).

The $w_d$ is combined with **$MAX_BUOY**, the range to the target and Seaglider's hydrodynamic model to calculate Seaglider's pitch angle on any given dive.

**Nominal Value:** 15

**Minimum Value:** 5

**Maximum Value:** 2880

### $T_GPS

**Definition:**

The maximum allowed time (minutes) to obtain a GPS position (GPS timeout). We often run with a **$T_GPS** of longer than 12.5 minutes, in order to ensure that the GPS receiver has time to receive a complete set of almanac entries in the event that lack of an almanac prevents getting a timely fix.

**Nominal Value:** 5

**Minimum Value:** 1

**Maximum Value:** 30

### $T_GPS_ALMANAC

**Definition:**

Time to wait (in minutes) to acquire the GPS almanac. The wait happens the next time the GPS is turned on. After the wait, the parameter resets to zero and the regular GPS operation (presumably a fix) will proceed. If the parameter is greater than zero, then the almanac sentences are checked every minute. The wait halts when the time has expired or at least ten satellites have recent almanac sentences. If the parameter is negative, then the wait only halts after the time has expired. A negative value also forces a complete NVRAM reset before the wait starts.

**Nominal Value:** 0

**Minimum Value:** -15

**Maximum Value:** 15

### $T_GPS_CHARGE

**Definition:**

Time to wait (seconds) before trickle-charging the GPS receiver (for Garmin GPS25 engines only). Negative values mean the GPS25 does not need charging. The GPS units now installed in Seagliders run on a button battery so there is no need to charge the system.

**Nominal Value:** -0.0033

### $T_LOITER

**Definition:**

The time (seconds) to loiter after going neutral at apogee, before pitching up and becoming positively buoyant for climb. While in the loiter state Seaglider will attempt to maintain zero vertical velocity. It will pump, but not bleed (become heavier) to do this. Seaglider does not servo on depth in this state. All other timeouts and depths are honored in this state. **$T_MISSION** and **$T_ABORT** need to be adjusted manually to account for the additional dive duration. Total dive duration is **$T_DIVE** + **$T_LOITER**. G&C and sampling intervals during the loiter state are controlled by the appropriate depth bins in the science file.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** 86400

### $T_MISSION

**Definition:**

The maximum mission time (minutes) allowed. After **$T_MISSION** divided by two, Seaglider transitions from dive phase to apogee phase, then commences the climb phase. If **$T_MISSION** is reached prior to Seaglider reaching **$D_SURF**, Seaglider immediately enters the surface phase. This time includes the dive, apogee and climb phases.

**Nominal Value:** 30

**Minimum Value:** 10

**Maximum Value:** 4320

### $T_NO_W

**Definition:**

The time (seconds) for Seaglider to wait with no significantly non-zero vertical velocity (less than 1 cm/s, as measured by dP/dt) before proceeding to the next phase of a dive. This is primarily used to move from the dive phase to the climb phase (initiate an apogee maneuver) when Seaglider unexpectedly encounters the bottom. Note that this protection is only in place at depths below **$D_FLARE** and **VBD** is already at **$MAX_BUOY** or below.

**Nominal Value:** 120

**Minimum Value:** 30

**Maximum Value:** 86400

### $T_RSLEEP

**Definition:**

The sleep time interval (minutes) during the recovery phase. During the recovery phase, Seaglider first gets a GPS fix, then calls the basestation up to **$CALL_TRIES** times to upload the GPS fix, then goes into low power sleep for **$T_RSLEEP** length of time. The surface evolution has about two minutes of "overhead," so that Seaglider calls are about (**$T_RSLEEP** + 2) minutes apart in practice.

**Nominal Value:** 3

**Minimum Value:** 0

**Maximum Value:** 14400

### $T_TURN

**Definition:**

The maximum amount of time (seconds) allowed to complete a turn during the active guidance and control (G&C) mode. If this timeout is reached before the heading is reached, Seaglider rolls back to neutral and continues.

**Nominal Value:** 225

**Maximum Value:** 10

**Maximum Value:** 720

### $T_TURN_SAMPINT

**Definition:**

The sample interval during active and passive G&C while turning. This should be short enough so that Seaglider cannot pass entirely through the heading deadband without sampling.

**Nominal Value:** 5

**Minimum Value:** 4

**Maximum Value:** 120

### $T_WATCHDOG (Set by manufacturer. Do not change.)

**Definition:**

The value to which the watchdog timer is set (minutes). This is an information only parameter so Seaglider's software knows the watchdog timer value. The watchdog timer itself is set with DIP-switches on the main board. If the watchdog timer expires, the main processor is reset, and Seaglider goes into recovery. This will not happen during normal operation and is meant as a fail safe against infinite loops in the software.

### $TCM_PITCH_OFFSET (Set by manufacturer. Do not change.)

**Definition:**

Static offset in pitch axis (degrees) between the compass output and the actual Seaglider body, as measured in the lab.

**Nominal Value:** 0

### $TCM_ROLL_OFFSET (Set by manufacturer. Do not change.)

**Definition:**

Static offset in roll axis (degrees) between the compass output and the actual Seaglider body, as measured in the lab.

**Nominal Value:** 0

### $TEL_NUM

**Definition:**

The telephone number Seaglider dials to connect to the basestation, 13 digits maximum. This is the PSTN number for the phone line connected to the first modem on a basestation for Seaglider operations. Format of the number: It starts with the international country code, without leading zeros (for example, "1" for the US), then city/area code and number. There are no spaces or other interrupting characters between the country code, city/area code or number.

If a communication session using the primary phone number (**$TEL_NUM**) does not successfully connect (after **$CALL_NDIVES** tries), the phone number is switched to the alternate number if available, **$ALT_TEL_NUM**) for the next surfacing. If a communication session completes successfully on the alternate phone number, the phone number is switched back to the primary for the next surfacing.

**Note:** This parameter is not adjustable from the cmdfile. The number can be edited either through the pdoscmds.bat file (see *Extended PicoDOS Reference Manual*, writenv on page 279) or through direct connection to Seaglider via the serial communications cable (see "Checking the Primary and Alternate Phone Numbers" on page 52).

### $TGT_AUTO_DEFAULT

**Definition:**

A Boolean parameter. If set to 1, automatically updates the default target in NVRAM. If set to 0, do not update the default target in NVRAM.

**Nominal Value:** 0

### $TGT_DEFAULT_LAT

**Definition:**

Floating point value (degrees decimal minutes) between -9000.000 and 9000.000. Together with **$TGT_DEFAULT_LON**, this parameter provides a default target location when the targets file cannot be read. For example, a latitude of 47 degrees 43.456 minutes would be input as 4743.456. Latitude values in the northern hemisphere will be positive values while latitude values in the southern hemisphere will be negative values.

**Nominal Value:** 4736.000

**Minimum Value:** -9000.00

**Maximum Value:** 9000.00

### $TGT_DEFAULT_LON

**Definition:**

Floating point value (degrees decimal minutes) between -18000.000 and 18000.000. Together with **$TGT_DEFAULT_LAT**, this parameter provides a default target location when the targets file cannot be read. For example, a longitude of -122 degrees 23.456 minutes would be input as -12223.456. Longitude values in the eastern hemisphere will be positive values while longitude values in the western hemisphere will be negative values.

**Nominal Value:** -12218.000

**Minimum Value:** -18000.000

**Maximum Value:** 18000.000

### $UNCOM_BLEED (Set by manufacturer. Do not change.)

**Definition:**

The uncommanded change in A/D counts of VBD bleed that triggers the following actions in an attempt to save Seaglider:

1. Stop whatever motor is running (the assumption is that electrical noise from one of the motors causes the Skinner valve to open) and disable it.
2. Close the Skinner valve
3. Enter the recovery state (go to the surface and call home).

**Nominal Value:** 50

**Minimum Value:** 0

**Maximum Value:** 400

### $UPLOAD_DIVES_MAX

**Definition:**

The maximum number of dives to upload at one surfacing. A value of -1 means upload all available dives that have not been previously uploaded.

**Nominal Value:** -1

**Minimum Value:** -1

**Maximum Value:** 9999

### $USE_BATHY

**Definition:**

If **$USE_BATHY** is -4, search for an on-board bathymap.nnn appropriate for the current position of Seaglider. This would be the standard usage in operating areas covered by more than one map. If **$USE_BATHY** is a positive integer then search for that particular on-board bathymap. If **$USE_BATHY** is 0 the feature is disabled and Seaglider either dives to **$D_TGT** or uses the on-board altimeter and its parameters to determine the appropriate depth at which to enter the apogee maneuver.

**Examples:**

If **$USE_BATHY** = -4, then search all on-board bathymap.nnn files for one that covers the current Seaglider position.

If **$USE_BATHY** = 0, then bathymetry maps are not used during the dive.

If **$USE_BATHY** = 7, then use the bathymetry file called bathymap.007 to get **$D_GRID**. No other map will be used.

**Nominal Value:** -4

**Minimum Value:** -4, 0

**Maximum Value:** 50

### $VBD_BLEED_AD_RATE

**Definition:**

The bleed rate in A/D counts per second that the code uses as the threshold for retries when bleeding. If the observed rate is less than this number, the bleed is stopped and restarted.

**Nominal Value:** 8

**Minimum Value:** 0

**Maximum Value:** 20

### $VBD_CNV (Set by manufacturer. Do not change.)

**Definition:**

VBD position conversion factor from A/D counts to $cm^3$. This is a constant determined by the geometry of the internal hydraulic fluid reservoir and the potentiometers. The sign is negative meaning that the higher A/D counts reflect more oil in the internal reservoir, hence a less inflated external bladder, hence a lower Seaglider displacement, and thus a lower Seaglider buoyancy.

**Nominal Value:** -0.2453

### $VBD_DBAND

**Definition:**

VBD position deadband ($cm^3$).

**Nominal Value:** 2

**Minimum Value:** 0

**Maximum Value:** 10

### $VBD_MAX (Set by manufacturer. Do not change.)

**Definition:**

Variable Buoyancy Device (VBD) position (A/D counts) software limit when the internal reservoir is almost full (external bladder fully bled, minimum Seaglider buoyancy). Seaglider's operating software closes the VBD main bleed valve (Skinner valve) when this value is reached. The builder or assembler typically sets this.

**Typical Value:**4000 (Value determined by the software maximum of the system. Obtain actual value from the vehicle's trim sheet.)

### $VBD_MAXERRORS

**Definition:**

Number of VBD errors allowed before entering recovery phase. This is a critical error parameter. Loss of VBD function can result in the loss of Seaglider!

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 5

### $VBD_MIN (Set by manufacturer. Do not change.)

**Definition:**

Variable Buoyancy Device (VBD) position (A/D counts) software limit when the internal reservoir is almost empty (external bladder fully pumped). Seaglider's operating software stops the VBD pump when this value is reached. The equipment manufacturer or assembler typically sets the VBD.

**Typical Value:**460 (Value determined by the software minimum of the system. Obtain actual value from the vehicle's trim sheet.)

### $VBD_PUMP_AD_RATE_APOGEE (Set by manufacturer. Do not change.)

**Definition:**

The pump rate in A/D counts per second that the code uses as the threshold for retries when pumping at apogee. If the observed rate is less than this number, the pump is stopped and restarted. The retries continue until the VBD timeout limit is reached, then an error is declared.

**Nominal Value:** 4

**Minimum Value:** 1

**Maximum Value:** 6

### $VBD_PUMP_AD_RATE_SURFACE (Set by manufacturer. Do not change.)

**Definition:**

The pump rate in A/D counts per second that the code uses as the threshold for retries when pumping at the surface. If the observed rate is less than this number, the pump is stopped and restarted. The retries continue until the pump timeout limit is reached, then an error is declared. If Seaglider is experiencing VBD retries at the surface, you may want to adjust the **$VBD_PUMP_AD_RATE_SURFACE** to 5.

**Nominal Value:** 6

**Minimum Value:** 1

**Maximum Value:** 8

### $VBD_TIMEOUT

**Definition:**

The total time (seconds) allowed for any commanded change in VBD position.

**Nominal Value:** 720

**Minimum Value:** 120

**Maximum Value:** 900

### $XPDR_DEVICE (Set by manufacturer. Do not change.)

**Definition:**

A configuration value specifying the model of the attached device, set by the manufacturer or assembler. These devices have dedicated hardware ports on all motherboard revisions and, as such, a port specification is not necessary. See **$COMPASS_DEVICE** on page 104.

**Nominal Value:** 24

**Minimum Value:** -1

**Maximum Value:** 1023

### $XPDR_INHIBIT

**Definition:**

A configuration value specifying the transponder inhibit time in hundreds of milliseconds. The inhibit time is the time after a transponder reply during which the transponder does not reply to subsequent interrogation. Shorter times mean the transponder can be interrogated more rapidly.

**Nominal Value:** 90

**Minimum Value:** 0

**Maximum Value:** 99 (9.9 seconds)

### $XPDR_VALID

**Definition:**

A configuration value specifying the transponder interrogation validation sensitivity in units of 0.5 ms. Valid values are from 0 (no validation) to 6 (3 ms). The validation value is the total time over a 10 ms window following initial triggering that the detector circuit must remain triggered. Longer validation times reduce spurious interrogation replies, but could result in decreased range. A value of zero results in no validation.

**Nominal Value:** 1

**Minimum Value:** 0

**Maximum Value:** 6

### $XPDR_PINGS

**Definition:**

This is an output from the glider, representing the count of the number of times the glider transducer responded to an external stimulus at its frequency setting during a dive. During missions this value should be zero. While an occasional ping is acceptable, frequent pings are not, as this activity consumes battery power. If excessive pings are seen in $XPDR_PINGS, tuning can be done using the $XPDR_INHIBIT and $XPDR_VALID parameters.

**Nominal Value:** 0

**Minimum Value:** 0

**Maximum Value:** No limit

CHAPTER 6

# *Pre-Deployment Tasks*

This chapter describes mission planning, prepping a Seaglider for transport to the field, pre-deployment self test and deployment. The following topics are covered:


- "Mission Planning" on page 160
- "Transporting Seaglider to the Field" on page 174
- "Final Launch Procedure" on page 180


To get the most out of a Seaglider deployment, both duration and data-wise, Seaglider must be properly prepped in the lab, self checks must be thoroughly reviewed and Seaglider's VBD, pitch and roll must be tuned when Seaglider is first deployed.

To achieve these goals a number of tasks must be completed by the pilot and the field team prior to releasing a Seaglider for a mission. These tasks include:

- Mission planning, including the modification of command, science and targets files
- Creating and loading of bathymetry files onto Seaglider, if desired
- Autonomous self test at or close to the deployment site and time
- Visual assessment, by the field team, of how Seaglider is riding in the water when it is first put in the water.

  - Approximately what angle is the nose pitch down in the water?
  - How much of the rudder or antenna is sticking out of the water?
- On land and in water testing of the transponder system

## *Mission Planning*

Mission planning is an important part of Seaglider piloting and the initial plan should be developed well before the vehicle is transported into the field for deployment. A basic understanding of Seaglider's operation, strengths and weaknesses is critical to planning effective science missions. The general idea is to go far and long by going slow--it's the square-law dependence of drag on velocity that gets you. "Half a knot on half a Watt." Seaglider motto.

The following sections give the operating limits of Seaglider.

### Environment

The range of stratification in which a Seaglider can operate normally is constrained by the total amount of VBD change available and the amount of (negative) buoyancy required for the flight plan. Pilots and/or scientists directing Seaglider's mission should determine the likely range of densities to be encountered on a proposed mission, and see if there is sufficient VBD range available to accommodate it. A typical Seaglider can operate over a density range of $10\sigma_T$. Compromises can be made by reducing maximum operating depth, at the expense of duration, or by reducing thrust at apogee, at the expense of horizontal speed.

The maximum sustained depth-averaged current that Seaglider can stem is 40 cm/s, or 0.8 knots. For a few individual dives it can go as fast as 50cm/s. That performance requires ballasting for 350cc of negative displacement, specifying vertical velocities of almost 20 cm/s. Dives to 1000m last about three hours in that case, and total mission length is on the order of six weeks. Remember that it is the average current over the full depth of the profile/dive that counts.

Surface currents can also be a problem, especially when doing shallow dives (see below). Plans for crossing strong currents, such as the Kuroshio or Gulf Stream, should be carefully considered, and contain both return (upstream) and bail-out plans.

## Endurance

Total endurance, is dependent on many factors, including depth of dive, vertical velocity, density stratification, and communications. The 24V lithium primary battery pack services the pump, mass shifter and the modem. The 10V lithium primary battery pack services the science sensors. Based on the way you're operating the vehicle either battery pack can be the limiting factor. When you're operating in shallow water resulting in frequent VBD pumping the 24V battery will likely run out first. When you're operating in deeper water doing very little science sensor sampling the 24V will still likely run out first. When you're operating in shallow or deep water and you're sampling the science sensors heavily, the 10V battery will run out first.

Seaglider has completed open-ocean missions more than nine months long, in conditions of small stratification (NE Pacific) where power conservation was the guiding factor. Missions north of Oahu (Hawaiian Ocean Time Series Station) typically lasted four months, due to stratification and the science requirement to resolve tides. Seaglider missions in the highly stratified Fjord -estuary of the Puget Sound are typically planned for only a few months.

Using the information gathered on the environment in the study area (depth, stratification, currents), as well as how long Seaglider is to be deployed, what horizontal distance it should cover in that time frame and what the scientific objectives are, determines overall mission length.

The VBD sub-system is the predominant energy consumer. Proper system trimming will allow for greatest energy efficiency.

Seaglider flies half a knot on half a watt (½ knot ~ 25 cm/s ~ 1 km/hr).

### *Depth and Efficiency*

Seaglider is least efficient operating in shallow water and most efficient in deep (up to 1000 m) water. The practical shallow water limit is about 50 m. It is hard to make progress toward a waypoint in water shallower than that, for three main reasons: turn radius, pump time, and surface time. Seaglider's turning radius (a few tens of meters at typical 25 cm/s horizontal speeds) is such that a significant portion of a shallow-water dive can be spent turning onto the desired course. Seaglider's standard buoyancy pump is optimized for efficiency at pressures equivalent to 1000 m ocean depth, its rate at shallow-water pressures (about 2 $cm^3/s$) means that a significant portion of a shallow-water dive can be spent pumping. Seaglider's

enhanced buoyancy pump is optimized for efficiency at pressures equivalent to <120m and 1000m ocean depth. By limiting the frequency with which the VBD pump operates, you conserve energy.

Finally, the time on the surface can be a significant percentage of the dive time, and if surface currents or winds are adverse, Seaglider can easily lose as much distance toward a waypoint while on the surface as it gains on the dive. Standard guidelines are to operate deeper than 200 m on offshore (deepwater) missions, and to try to stay deeper than 50 m on coastal or estuarine missions.

### *Uncontrollable Environmental Factors*

**Stratification**

The number of density layers and their magnitude in the water column will affect Seagliders power consumption. The more density layers and the larger the density change present that Seaglider has to overcome, the more energy it will use when pumping to achieve the needed buoyancy.

**Temperature**

Operating in colder environments can reduce the total amount of energy available from the batteries.

**Ocean Currents**

An environment with strong ocean currents (surface and or subsurface) will necessitate more thrust which is supplied by more pumping to make headway toward desired waypoints. This is true only if the current is in a direction that opposes the desired heading.

### *Pilot Controllable Factors*

**Seaglider Trimming**

A well trimmed Seaglider will require less corrective action while in flight and conserve both the 10V and 24V power supplies. The primary piloting parameters below will have the greatest impact on power consumption.

### $D_TGT + $T_DIVE

The ratio of $D_TGT and $T_DIVE controls the speed at which the Seaglider dives and climbs. Adjusting the ratio to perform slower dives results in longer dive time and therefore increases the amount of time between apogee and surface VBD pumping maneuvers giving more operational time relative to energy use.

### $MAX_BUOY

Decreasing the amount of oil used to for thrust decreases the overall VBD usage and conserves the energy stored in the 24V battery.

### $SM_CC

Decreasing the amount of oil used for the surface maneuver decreases the overall VBD usage and conserves the energy stored in the 24V battery.

### $CALL_N_DIVES

Skipping Iridium transmissions during specified dives conserves the energy stored in the 24V battery.

### $CAPUPLOAD

Turning off the transmission of the .cap file conserves the energy stored in the 24V battery. This file, which contains a detailed record of the entire dive, can be quite large, causing the glider to remain at the surface for an extended period of time while its contents are transferred to the Basestation via Iridium.

### G&C interval

Increasing the G&C interval value (in the science file) decreases the frequency at which the guidance and control is checked conserving the energy stored in the 10V battery resulting in fewer attitude and buoyancy corrections which conserves the energy in the 24V battery.

#### Science file

Increasing the number of seconds between science sensor samples will decrease the frequency at which the sensors are powered on and conserve the energy stored in the 10V battery.

**Remove all unneeded commands**

Removing all unneeded command files from the basestation (targets, science, pdoscmds.bat …) will decrease the amount of data transferred and conserve the energy stored in the 24V battery.

For mission planning on deep dives, we recommend using the below table to estimate energy usage.

**Energy Budget for a 1000-Meter Dive**

| Seaglider Subsystem | Energy Consumed | Percent of Battery Used |
|---|---|---|
| Pump* | -9kJ | 70% |
| Computer | -2kJ | 15% |
| Instruments | -1kJ | 7% |
| Telemetry | -1kJ | 7% |
| Total | -13kJ | 100% |

* (For a deep, slow dive of 200cc of thrust and stratification of 150cc)

### *MIssion Duration Examples*

Below are some examples of expected endurance based on some conditional assumptions.

**Base conditions for all dives**

MAX_BUOY=150

SM_CC=350

Calls per dive = 1.5

10 cm/sec vertical velocity

5% of packs used at beginning of mission

5% of packs left at end of mission

No capture files being transmitted

Call in after every single dive

Above 100m sample dissolved oxygen, WetLabs, and Conductivity & Temperature Sail every 10 seconds, Guidance & Control Interval = 60.

Between 100m-500m sample Conductivity & Temperature Sail every 10 seconds, and WetLabs and Dissolved Oxygen every 30 seconds, Guidance & Control interval = 180.

Between 500m-1000m sample Conductivity & Temperature every 10 seconds, and WetLabs and Dissolved Oxygen every 60 seconds, Guidance & Control interval = 180.

| Sensors | Buoyancy Engine Option | 100m Duration (Days) | 100m Limiter | 500m Duration (days) | 500m Limiter | 1000m Duration (days) | 1000m Limiter |
|---|---|---|---|---|---|---|---|
| Seabird Conductivity & Temperature Sail, WetLabs Sensor | Standard Buoyancy Engine | 55 | 24V | 124 | 10V | 206 | 10V |
| Seabird Conductivity & Temperature Sail, WetLabs Sensor | Enhanced Buoyancy Engine | 69 | 24V | 121 | 24V | 184 | 24V |
| Seabird Conductivity & Temperature Sail, Seabird Disolved Oxygen Sensor | Standard Buoyancy Engine | 55 | 24V | 153 | 10V & 24V | 221 | 24V |
| Seabird Conductivity & Temperature Sail, Seabird Disolved Oxygen Sensor | Enhanced Buoyancy Engine | 69 | 24V | 121 | 24V | 183 | 24V |

| Sensors | Buoyancy Engine Option | 100m Duration (Days) | 100m Limiter | 500m Duration (days) | 500m Limiter | 1000m Duration (days) | 1000m Limiter |
|---|---|---|---|---|---|---|---|
| Glider Payload Conductivity, Temperature & Density Sensor, Wet-Labs Sensor | Standard Buoyancy Engine | 54 | 24V | 96 | 10V | 118 | 10V |
| Glider Payload Conductivity, Temperature & Density Sensor, Wet-Labs Sensor | Enhanced Buoyancy Engine | 68 | 24V | 96 | 10V | 118 | 10V |

### cmdfile Edits

The starting point for making edits to the cmdfile is the log files generated by the simulated dives. The log files list all parameters and the value assigned to each.

1. Open the last simulated dive's log file and using the "Alphabetized Parameters" on page 92, go through ***every*** parameter and see if all are set correctly for the upcoming mission.

   For example, the initial dive should be relatively shallow (~45m) to check the initial positions of system centerpoints. So **$D_TGT** should be set to **$D_TGT,45**. Correspondingly, the desired vertical velocity of the initial dives is usually 10cm/s so **$T_DIVE** should be set to **$T_DIVE,15**. **$T_MISSION** should be set to **$T_MISSION,25**.

   If Seaglider is to fly from waypoint to waypoint rather than by heading, **$HEADING** should be **$HEADING,-1**. **$MAX_BUOY** and **$SM_CC** settings are environment dependent and should initially be set conservatively. **$MAX_BUOY,150** and **$SM_CC** near maximum.
2. Check **$C_VBD**, **$C_PITCH**, **$C_ROLL_DIVE** and **$C_ROLL_CLIMB** values against those listed in the Cal sheet of the notebook that came with Seaglider.
3. Turn on the **$USE_BATHY** feature if bathmetric maps are to be used.
4. Turn on the Kalman filter, if that feature is desired.
5. Check that **$T_RSLEEP,3** has been set.

6. Make sure that the last line of the command file is initially **$QUIT.**

   This holds Seaglider at the surface when it is first put in the water for the deployment. If the command **$RESUME** is inadvertently left in place, Seaglider will "dive" as soon as it is given permission to launch and you will not know exactly what state it will be in when it gets in the water. Seaglider needs to remain at the surface when it is first put in the water so that the transponder and communication checks can be completed and how Seaglider is sitting in the water can be evaluated.

### targets Edits

Next, determine what path Seaglider should fly during its mission. Again, this can be edited as the mission progresses. When the waypoints have been determined, edit the targets file with this information. An example of a targets file and a description of each column is below.

```
/ Sample North Carolina Coast targets
NCSPONE     lat=3357.4     lon=-7623.5     radius=200     goto=WPONE
WPONE       lat=3356.0     lon=-7625.2     radius=200     goto=WPTWO
WPTWO       lat=3353.9     lon=-7625.2     radius=200     goto=WPTHREE
WPTHREE     lat=3352.2     lon=-7621.4     radius=200     goto=WPFOUR
WPFOUR      lat=3356.6     lon=-7623.8     radius=200     goto=NCSPONE
```

- Column 1: Name of the waypoint
- Column 2: Latitude of waypoint in degrees, minutes, decimal minutes (no spaces between degrees and minutes)
- Column 3: Longitude of waypoint in degrees, minutes, decimal minutes
- Column 4: Distance in meters Seaglider can be from the waypoint and still considere having reached the waypoint
- Column 5: Name of the next waypoint to go to after it has reached the present waypoint

For new missions, Seaglider heads to the first waypoint listed, in this case the one called NCSPONE. To change the waypoints file during a mission the pilot can upload a new waypoints file. However, use caution when doing this. If an active target name from the old targets file is in the new targets file, Seaglider retains that active target. If an active target is not in the new targets file, then Seaglider goes to the first target in the list.

If the pilot wants to change the waypoint, Seaglider is going to in the targets file that is loaded on Seaglider (for example, Seaglider is heading to NCSPONE in the example above and the pilot wants Seaglider to head to WPTWO instead) the pilot sends this change to Seaglider via the pdoscmds.bat file. See Appendix C, “Extended PicoDOS® Reference Manual, v66.07” on page 303 for the correct syntax to make the change.

If there is no valid targets file on Seaglider, it uses the default target (**$TGT_AUTO_DEFAULT, $TGT_DEFAULT_LAT, $TGT_DEFAULT_LON**). It is highly recommended that you enter the latitude and longitude coordinates of a location as DEFAULT where you can recover Seaglider in the event that it loses or cannot read its targets file.

### Science Sensors

Seaglider supports two science sensor interfaces to allow a variety of sensors to be connected to the vehicle via the serial ports. The two interfaces, Ordinary Serial Interface (OSI) and Autonomous Logger Interface (ALI), are described below.

#### *Ordinary Serial Interface (OSI)*

OSI is a .cnf file configurable serial sensor interface that allows new instruments to be added to the glider without writing new binary drivers or modifying source code. OSI is used exclusively for sensors without the ability to log data to an internal file system.

Note that OSI devices can be configured either through the provided .cnf files or via pre-established options available in the configuration menus described in Appendix F.

iRobot currently provides .cnf files for the following OSI sensors: Biospherical PAR, WET Labs (BB2F, B2FL, BFL2)

#### *Autonomous Logger Interface (ALI)*

ALI is a .cnf file configurable serial sensor interface that allows new instruments to be added to the glider without writing new binary drivers or modifying source code. ALI is used exclusively for sensors with the ability to log data to an internal file system.

iRobot currently provides .cnf files for the following ALI sensor: Sea-Bird GPCTD

All Autonomous Logger Interface (ALI) devices recognize five cmdfile parameters: xx = the prefix for a given sensor. Example for the GPCTD, the prefix = PC

### science Edits

Control of the sampling and depth intervals for the OSI sensors, to meet the science requirements of the mision, are specified in the science file. Note that ALI sensors (described in the previous section) are not controlled via the science file. For information on controlling ALI sensors see Chapter 9, “Files for Operations” on page 237.

The practical lower limit on sampling is 4 seconds. If only the conductivity and temperature sensors are sampled, it is possible to sample every 4 seconds, but with the oxygen and BBFL2 or BB2FL optical sensors also being sampled, 5 seconds is the lower limit. The science file also provides the ability to turn off sensors, or only energize them every *nth* sample, in a given depth range (or ranges). An example science file is below.

```
// Science for North Carolina Coast
/for Seaglider w sensors/: CT, SBE-43F oxy, WET Labs BB2FL-VMT
/depth      time    sample    gcint
50          5       111       30
200         5       121       60
300         5       103       120
```

- Column 1: The bottom of the depth bin in meters for that sampling protocol
- Column 2: Base sampling interval in seconds
- Column 3: Multipliers for each of the three sensors listed on line 2 of the file, in the order listed, applied to the base sampling interval
- Column 4: Guidance and Control (G&C) sampling interval in seconds

The science sampling works as follows:

For the first row of sampling protocol, sampling in depths from 0-50m, each of the sensors is sampled once every 5 seconds – multiply the 5 in column 2 by the first digit (1) in column 3 for CT, by the second digit (1) in column 3 for SBE-43F and by the third digit (1) in column 3 for the WET Labs sensor. G&C sampling between 0 and 50m occurs once every 30 seconds.

Sampling Rate = Sample time x respective sensor sample interval

Where:

- Sample time = value in column 2
- CT interval = 1st digit of the value in column 3
- SBE-43 interval = 2nd digit of value in column 3
- WET Labs interval = 3rd digit of value in column 3

For the second row of sampling protocol, depths between 50 and 200m, the CT and WET Labs sensors are sampled once every 5 seconds – multiply the 5 in column 2 by the first digit in column three for the CT sensor and multiply the 5 in column 2 by the third digit in column three for the WET Labs sensor. The SBE-43F oxygen sensor is sampled once every 10 seconds – multiply the 5 in column 2 by the second digit in column 3. The G&C sampling occurs once every 60 seconds.

Point to point variability may be seen in the WET Labs data due to the differences in sampling frequency between the WET Labs puck and the glider. The WET Labs puck samples at 1Hz (one sample/second). The default sample averaging by the puck is also once/second. However, the glider sampling frequency is no faster than 1/4Hz (once every 4 seconds) and is dependent on the number of sensors being sampled. The frequency that the glider interrogates the WET Labs puck, as well as any other science sensor, for a data point is set by the user in the science file and is often between 1/5Hz (once every 5 seconds) and 1/10Hz (once every 10 seconds). This means that only 1/5th to 1/10th of the WET Labs data samples are recorded by the glider which can accentuate any point to point variation in the WET Labs data. To decrease the point to point variation seen in the WET Labs data, the user can increase the averaging interval for the WET Labs sensor to 2 seconds.

For depths below 300m, Seaglider continues to use this sampling routine. Note that the bound on depth interval is dependent on pressure sensor sampling and Seaglider vertical velocity. The pressure is only sampled at the base interval, so if that is large and Seaglider is going fast, the switch to the next regime may be deeper/shallower than expected. Choices made in the science sampling have an impact on the energy use of Seaglider, especially the 10V battery pack. More samples and shorter G&C intervals use more processor and sensor power than do fewer samples and longer G&C intervals.

### Bathymetric Files

Map files provide Seaglider with geographic environmental information, specifically bathymetry about a given region of the ocean. Seaglider can carry up to 999 bathymetry maps (the files are named bathymap.nnn, where .nnn is the map number), but in practice far fewer are typically on board. These maps are not required for Seagliders to fly, but provide a low energy means to determine apogee depth. Note that the values in the bathymap files represent the depth at which Seaglider starts its apogee maneuver, NOT the actual bottom depth.

Bathymap files of the mission area can be generated from several sources. iRobot is obtaining the files used in its gliders from the University of Washington's web site:

http://iop.apl.washington.edu/Seaglider/bathy.php

The files generated are zipped. They must be unzipped and named following the bathymap.nnn protocol.

After the files are generated, load them onto Seaglider's flashcard following the procedure below:

1. Connect the laptop to Seaglider via the communication cable.
2. Turn on the laptop and start a screen capture or log file.
3. Wand on Seaglider.
4. Press ENTER when Seaglider output appears on the laptop.
5. Set the date and time.
6. Select appropriate power source.
7. Type **4** (pdos) from the Main Menu, and then press ENTER.
8. At the PicoDOS prompt, type:

   **xr bathymap.xxx**

   where xxx is the number (from 001-999) of the bathymap.
9. From the terminal emulator (for example, Tera Term) menu, choose File>Transfer>XMODEM>Send.
10. Select the desire bathymap file.

    The bathymap file is downloaded to Seaglider.

    The watchdog timer may cause Seaglider to reboot if more than 10 minutes are spent in PicoDOS. If this happens and you are not finished loading bathymap files repeat steps 8-10.
11. To exit PicoDOS, type **quit** and press ENTER.

### .pagers file

For directions on setting up a .pagers file, see Chapter 7, “Pre-Launch Procedures” on page 187.

### .mailer file

This file controls the sending of data products created during basestation processing via email. Mail is formatted as a MIME document and sent via SMTP

Each line of this file is of the format

smtp_mail_address[,body|msgperfile|kkyy_subject|gzip] [,[eng| log| pro| bpo| csv| asc| cap| comm| dn_kkyy| up_kkyy| nc| mission_ts| mission_pro]+] | ,all

where:

smtp_mail_address - a valid smtp mail address - gliderpilot@apl.washington.edu

For example:

body - send files in the body of the message, instead as attachments (default).

Incompatible with gzip, nc, mission_ts and mission_pro

msgperfile - send each file in its own message, instead of sending all files that have been

newly created in a single message (default)

kkyy_subject - Use the navy specified 'XBTDATA' subject line, instead of a more

informative subject (default)

gzip - compress all files before sending

eng, log, pro, bpo, csv, asc, cap , dn_kkyy, up_kkyy, nc - send any newly created files of the specified extension

comm - send the comm.log file

mission_ts, mission_pro - send the mission timeseries or mission profile, if they have been updated on the processing

all - send all the newly created or updated files

**Examples**

- Send NAVO kkyy files

#kkyyuser@navo.navy.mil,msgperfile,body,kkyy_subject,dn_kkyy,up_kkyy

- Send out the typical per-dive files

#someone@apl.washington.edu,gzip,log,eng,cap,nc,comm

- Send a collaborator the update mission profile (be careful, this can be a big file)

#someone@u.washington.edu,gzip,mission_pro

### .URL file

These are URLs to GET for each processed dive. The urls are called twice during processing - the first time, after all the per-dive files have been processed - in which case Base.py adds the arguments:

'instrument_name=sg<xxx>&dive=<dive>&files=perdive'

The second time is at the end of processing, after all the whole mission files have been generated, in which case Base.py add the arguments:

'instrument_name=sg<xxx>&dive=<dive>&files=all'

First entry on the line is the timeout to wait for a response to the GET

It is separated by a whitespace from the URL

Comments in the file are indicated by '#'

**Example**

1 http://mydomain.edu/~glider/cgi-bin/update.cgi

## *Transporting Seaglider to the Field*

When Seaglider's deployment time is close, transport Seaglider to the field. If it is a short trip, Seaglider may be able to be transported in its cradle. Remove the antenna from the aft end of Seaglider, leaving the cables attached (see below). It may also be necessary to remove the wings.

If getting Seaglider to the field involves a longer trip and/or commercial shipping, put Seaglider in **Travel Mode** and use the shipping crate.

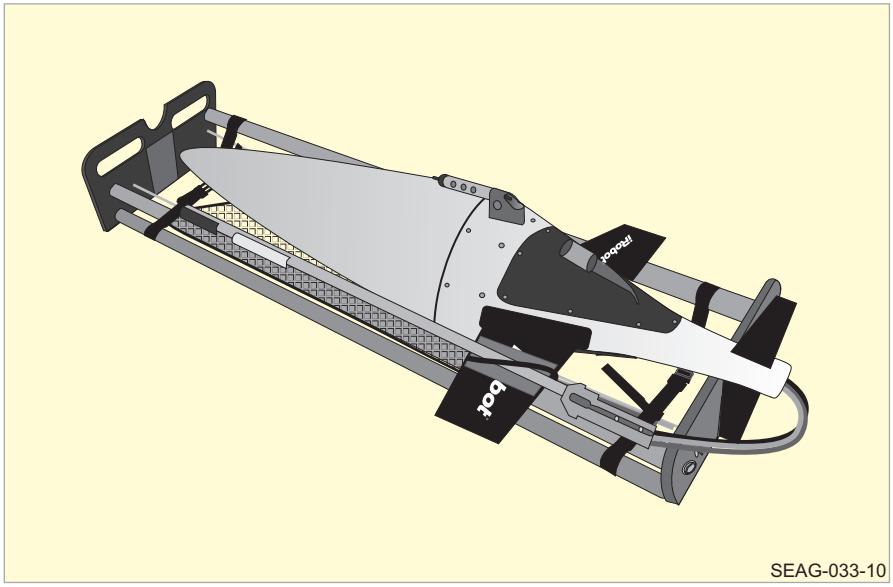
### Detaching the Antenna Mast

To detach the antenna mast from the aft fairing:

1. Remove the rudder screws.
2. Slide the antenna mast and rudder boot out of the aft fairing (can best be facilitated if the rudder section is outside of the cradle as in Figure 6-1 on page 175).
3. Fold the antenna mast back onto Seaglider's wing and secure it with protective foam and a bungee cord.
4. Leave the rudder in its aft fairing slot and put in the rudder screws to hold it in place.

   Figure 6-1 shows Seaglider with the antenna mast removed.

**FIGURE 6-1. Seaglider with Antenna Mast Removed from the Aft Fairing**




## Putting Seaglider in Travel Mode

When you put Seaglider in travel mode, you bleed most of the hydraulic oil from the external bladder to the internal reservoir, making it safe for travel, and move the battery to its most stable position. Seaglider does not have to be outside, nor does the antenna have to be fully installed.

To put Seaglider in travel mode:

1. Connect Seaglider to the laptop via the communication cable.
2. Turn on the laptop and start the terminal emulation program and a screen capture or log file.
3. Wand on Seaglider. When output from Seaglider appears on the computer screen press ENTER within one minute.
4. Accept the default when prompted for DATE and TIME.
5. When queried “Are you running on external (bench) power?” answer Y or N, depending upon how Seaglider is being powered.
6. Go to the Main Menu seen in Figure 6-2.
7. At the Main Menu, select 2: Hardware tests & Monitoring and press ENTER.

**FIGURE 6-2. Travel Mode Menu -1**

```
------ Hardware Menu ------
*Motors and VBD
     1 [pitch   ]  Pitch control
     2 [roll    ]  Roll control
     3 [vbd     ]  VBD control
     4 [pressure]  Pressure sensor
     5 [compass ]  Compass (tcm2)
     6 [gps     ]  GPS
     7 [modem   ]  Modem (xmodem mode)
     8 [intpress]  Internal pressure
     9 [altim   ]  Altimeter
*Sensors
    10 [ct      ]  C,T sensors
    11 [nullct  ]  NULL C,T sensors
    12 [o2      ]  O2 sensor
    13 [wetlabs ]  WETLabs VSF & fluorometer
    14 [par     ]  QSP 2150 PAR sensor
    15 [optode  ]  Optode
*Other
    16 [batt    ]  Batteries and fuel gauges
    17 [lowlevel]  Low-level hardware (IO,A-D,CF)
    18 [misc    ]  Miscellaneous (travel, timeouts, date/time)
    19 [develop ]  Developer tests
    CR)  Return to previous
Enter selection (1-19,CR): 18   ———————— Step 8
```

8. Type **18** ([misc] Miscellaneous (travel, timeouts, date/time)), and press ENTER.

9. When **Miscellaneous Hardware Functions** appears, type **1** ([Travel] Prepare for travel (Figure 6-3)). Seaglider then bleeds the oil from the external bladder into the internal resevoir.

**FIGURE 6-3. Travel Mode Menu – 2**

```
------ Miscellaneous hardware functions ------
     1 [travel  ]  Prepare for travel
     2 [iotime  ]  Change user IO timeout
     3 [timeout ]  Change watchdog time (mins)
     4 [date    ]  Read/set time-of-day
     5 [zerost  ]  Reset self-test counter to zero
     6 [fault   ]  Force a fault - WARNING, CRASHES to TOM8!!!!
     7 [dump    ]  Dump captured registers
    CR)  Return to previous
Enter selection (1-7,CR): 1 ———————————— Step 9
Bleeding down...
1314.122,HVBD,N,Bleed commanded from 405.96 cc (1755) to 15.21 cc (3
1315.321,HVBD,N,403.5 cc (ad: 1765 [1772, 1771]) ...................
1326.521,HVBD,N,336.1 cc (ad: 2040 [2027, 2065]) ...................
1337.796,HVBD,N,271.5 cc (ad: 2303 [2304, 2310]) .......
```

10. When the “good to go!” message appears, wand Seaglider off. Make sure that Seaglider is off by repeatedly pressing ENTER on the laptop keyboard. If there is no response, Seaglider is turned off.

### Packing Seaglider in its Shipping Case

To pack Seaglider in its shipping case:

1. Remove the screws holding the rudder in place and remove the rudder. Store the screws in the spare parts kit. Place the rudder in its compartment in the shipping case.
2. Slide the antenna mast and rudder boot out of the aft fairing (can best be facilitated if the rudder section is outside of the cradle as in Figure 6-1 on page 175).
3. Fold the antenna mast back onto Seaglider’s wing and secure it with protective foam and a bungee cord.
4. Remove the wings.
5. Store the screws in the spare parts kit, being sure to secure the antenna in its slots alongside Seaglider.
6. Place the wings in their compartment in the shipping case.
7. Put Seaglider in the shipping crate, securing the antenna.

8. Put the laptop, communication cable, notebook, spare parts kit and anything else that originally came out of the shipping case back in it.

### Shipping Seaglider

Please be advised that lithium battery shipments are controlled by the Department of Transportation (DOT), International Civil Aviation Organization (ICAO), and the International Air Transport Association (IATA). Under the US DOT regulations, please review requirements under 49 CFR 172.101 and Special Provisions 29, 188, 189, 190, A54, A55, A101, and A104 and packing instruction 49 CFR 173.185. Under the IATA regulations, please review 4.2 List of Dangerous Goods and Special Provision A48, A88, A99, A154, A164 and Packing Instruction 968, 969, or 970.

### Assembling Seaglider

Re-assemble Seaglider, following the instructions in “Assembling Seaglider” on page 41. Follow the directions for installing the antenna mast on page 42.

### Performing the Interactive Self Test

Run the interactive self test “Interactive Pre-launch Self Test” on page 188. This can be done on shore or at sea. When Seaglider calls into the basestation, it should pick up the cmdfile, science and targets files that you edited (see “cmdfile Edits” on page 166, “targets Edits” on page 167, and “science Edits” on page 169.

At the conclusion of the self test, if it passed, the field team stands by, laptop still connected to Seaglider, while the pilot reviews all of the files that were generated and makes sure that all systems are functioning as expected giving permission to start the Sea Launch sequence. The field team needs to be patient as this process can take 15 minutes. The pilot needs to review the following:

**.cap** file (ptnnnxxxx.cap; where p=processed, t=self test, nnn=Seaglider number, xxxx=self test number)

- Look for errors and warnings.
- Were file transfers successful?
- Was a GPS fix obtained?
- Did the motors move as expected?

**.eng** file (ptnnnxxxx.eng; where p=processed, t=self test, nnn=Seaglider number, xxxx=self test number)

- Check the science data as best as possible.
- Does the file header line match the sensors that are installed?
- Are the temperature values from the CT sensor realistic?
- Are the conductivity values very close to zero?
- There should be some oxygen trapped in the DO sensor. Is there a dissolved oxygen reading greater than zero?
- For the optical sensors, are the values of counts greater than zero and in the expected data columns?
- Are the reference values the expected reference value and are they in the correct columns?

**.log** file (ptnnnxxxx.log; where p=processed, t=self test, nnn=Seaglider number, xxxx=self test number)

- The list of parameters and the value assigned to each.
- Do the parameter values look okay?

.**pvt** file (ptnnnxxxx.pvt; where p=processed, t=self test, nnn=Seaglider number, xxxx=self test number)

- Check that Seaglider's ID, password and phone numbers are as expected.

If all of the data in the files looks good the pilot gives the field team permission to Launch.

### Communication Messages

When attempting to make a phone call with the Seaglider, it is very common to see different types of error messages returned. Below are some commonly returned messages along with their meaning.

**NO CARRIER** - Phone call has dropped. The modem returns this string when a call has ended.

**REGISTRATION CHECK FAILED** - The AT+CREG? command has returned that it is not registered on the satellite network. This is not a fatal error; the glider will continue with operations, skipping the geo-registration. If the Seaglider is still able to make a phone call, this is not a major issue; in order to make a phone call, you must be registered on the satellite network.

**UNABLE TO LOGIN** - There was a problem logging into the basestation. This could be due to corruption in the communications channel or a phone call that dropped prematurely.

**NO REPSONSE TO INITIAL AT ATTEMTS** - The Seaglider determines if the modem is awake and powered on by sending "AT" and then expecting "OK" in response. If it does not receive the AT, a failure occurs. This can indicate a modem is exhibiting faulty behavior.

**NO LOGIN: PROMPT DETECTED** - See unable to login above.

**BASESTATION NOT READY TO RECIEVE** - See unable to login above (i.e., corruption in the comms channel or a dropped call.

## *Final Launch Procedure*

If the pilot has given the field team permission to start a SeaLaunch, the field team needs to:

1. At the Main Menu, type **5** (Pre-Launch), and then press ENTER.

**FIGURE 6-4. Pre-Launch**

```
------ Main Menu ------
     1 [param    ]  Parameters and configuration
     2 [hw       ]  Hardware tests and monitoring
     3 [modes    ]  Test operation modes and files
     4 [pdos     ]  PicoDOS commands (and exit)
     5 [launch   ]  Pre-launch
Enter selection (1-5,CR):
```

2. Type **7** (Sea Launch!), and then press ENTER.
3. At the prompt, "Can the antenna be used for GPS and communications? [Y]," press ENTER to accept the default (Y).

   Seaglider then resets **$SIM_PITCH** and **$SIM_W** to 0, and removes old data and log files from the persistor CF2 flashcard.

**FIGURE 6-5. Sea Launch**

```
------ Launch Menu ------
     1 [scene   ]  Set scenario mode
     2 [selftest]  Perform interactive self test
     3 [autotest]  Perform autonomous self test
     4 [uploadst]  Upload self-test results
     5 [reset   ]  Reset dive/run number
     6 [test    ]  Test Launch!
     7 [sea     ]  Sea Launch!
    CR)  Return to previous
Enter selection (1-7,CR): 7 ———————— Step 2

Can the antenna be used for GPS and communications? [Y] y ——— Step 3
1161.831,SPOWER,N,All devices are off.
1161.882,SUSR,N,Resetting $SIM_W and $SIM_PITCH to 0.0
1161.952,HTT8,N,Updating parameter $SIM_W to 0
1162.013,HTT8,N,Updating parameter $SIM_PITCH to 0
1162.103,HTT8,N,Writing NVRAM...done.
1169.577,SUSR,N,Removing old data and log files
Deleting ST0032DU.X00...
Deleting ST0032KZ.X...
Deleting ST0032LU.X00...
Deleting ST0032LU.X01...
```

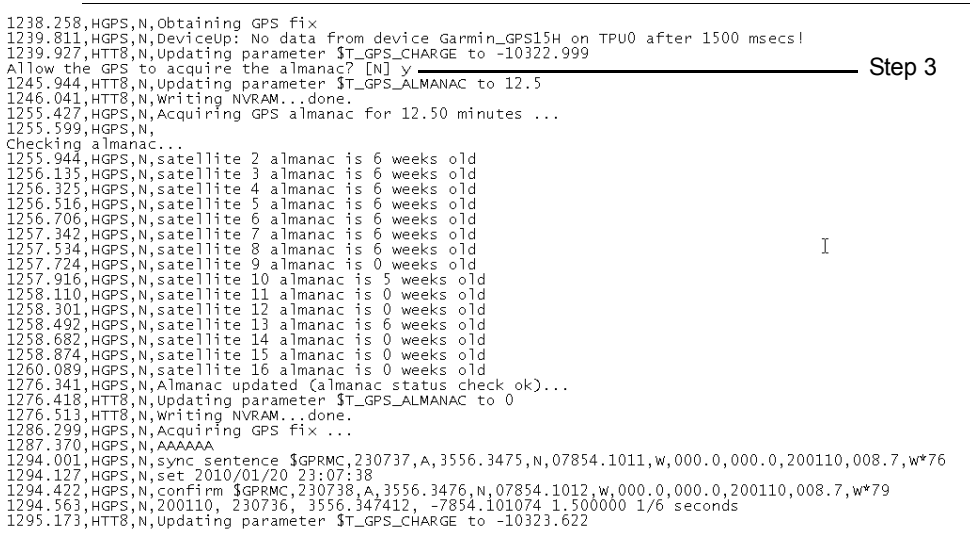
4. At the prompt, “New telnum? [value],” if the value shown is for the current telnum is correct (check with pilot if necessary), press ENTER. Otherwise, type a *value* and press ENTER. See page 199 for the correct phone number format.
5. At the prompt, “New altnum? [value],” if the value shown is for the current altnum is correct (check with pilot if necessary), press ENTER. Otherwise, type a *value* and press ENTER. See page 199 for the correct phone number format.
6. Seaglider tries to obtain a gps fix. The message “DeviceUp: No data from device Garmin_GPS 15H on TPUO after 1500 msecs!” is to be expected.The GPS unit on average needs 4 seconds after startup to obtain a fix and output data. The value 1500 msecs is hard coded into the software and changing it requires a re-compile of the code. The response time will be increased from 1500 msec on the next code release.

   Seaglider prompts “Allow the GPS to acquire the almanac? [N].”
7. **For Seagliders with ID numbers 515 and lower:** Type **Y** and then press ENTER. Seaglider acquires the almanac, writes the results to NVRAM and obtains a gps fix. Seaglider obtains a GPS fix, updates the almanacs and synchronizes the time onboard Seaglider.

   **For Seagliders with ID numbers 516 and higher:** The Garmin next generation GPS unit is installed. It has the same functionality as the unit installed in

earlier Seaglider units, however, now the firmware automatically acquires the almanacs. Therefore the user when prompted, “Allow the GPS to acquire the almanac? [N],” should press ENTER to accept the default ‘No.’ Should the user type ‘Y’ the output looks very different from the output of the earlier model to the point of looking erroneous. If the unit is successfully acquiring GPS fixes it is functioning properly.

**FIGURE 6-6. GPS Fix**


1238.258,HGPS,N,Obtaining GPS fix
1239.811,HGPS,N,DeviceUp: No data from device Garmin_GPS15H on TPU0 after 1500 msecs!
1239.927,HTT8,N,Updating parameter $T_GPS_CHARGE to -10322.999
Allow the GPS to acquire the almanac? [N] y
Step 3
1245.944,HTT8,N,Updating parameter $T_GPS_ALMANAC to 12.5
1246.041,HTT8,N,Writing NVRAM...done.
1255.427,HGPS,N,Acquiring GPS almanac for 12.50 minutes ...
1255.599,HGPS,N,
Checking almanac...
1255.944,HGPS,N,satellite 2 almanac is 6 weeks old
1256.135,HGPS,N,satellite 3 almanac is 6 weeks old
1256.325,HGPS,N,satellite 4 almanac is 6 weeks old
1256.516,HGPS,N,satellite 5 almanac is 6 weeks old
1256.706,HGPS,N,satellite 6 almanac is 6 weeks old
1257.342,HGPS,N,satellite 7 almanac is 6 weeks old
1257.534,HGPS,N,satellite 8 almanac is 6 weeks old
1257.724,HGPS,N,satellite 9 almanac is 0 weeks old
1257.916,HGPS,N,satellite 10 almanac is 5 weeks old
1258.110,HGPS,N,satellite 11 almanac is 0 weeks old
1258.301,HGPS,N,satellite 12 almanac is 0 weeks old
1258.492,HGPS,N,satellite 13 almanac is 6 weeks old
1258.682,HGPS,N,satellite 14 almanac is 0 weeks old
1258.874,HGPS,N,satellite 15 almanac is 0 weeks old
1260.089,HGPS,N,satellite 16 almanac is 0 weeks old
1276.341,HGPS,N,Almanac updated (almanac status check ok)...
1276.418,HTT8,N,Updating parameter $T_GPS_ALMANAC to 0
1276.513,HTT8,N,Writing NVRAM...done.
1286.299,HGPS,N,Acquiring GPS fix ...
1287.370,HGPS,N,AAAAAA
1294.001,HGPS,N,sync sentence $GPRMC,230737,A,3556.3475,N,07854.1011,W,000.0,000.0,200110,008.7,W*76
1294.127,HGPS,N,set 2010/01/20 23:07:38
1294.422,HGPS,N,confirm $GPRMC,230738,A,3556.3476,N,07854.1012,W,000.0,000.0,200110,008.7,W*79
1294.563,HGPS,N,200110, 230736, 3556.347412, -7854.101074 1.500000 1/6 seconds
1295.173,HTT8,N,Updating parameter $T_GPS_CHARGE to -10323.622

8. Seaglider calls the basestation and uploads the cmdfile and if present the science, targets and pdoscmds.bat files, and downloads the files generated by the sea launch routine.
9. Make sure that the directive **$QUIT** is the last line of the cmdfile.

   **Field team:**

   Seaglider gets a GPS fix, then call the basestation. After it logs out, it asks the operator, “Do you have permission to launch?”

   - Call the pilot.
   - Standby while the pilot checks the pnnn000.prm (p=processed, nnn=Seaglider number, 0000=dive 0) file. The field team cannot launch until the pilot grants the permission to launch!

**Pilot:**

- The pilot MUST check the .prm file completely before continuing. The .prm file starts with a section captured from the end of the self test. You can see the statement the field team saw on their laptop connected to Seaglider. Next, it shows a list of every parameter and the value assigned to it. Are ALL of the parameters set correctly? Were **$SIM_PITCH** and **$SIM_W** reset to 0?
- If all looks good to the pilot, the pilot tells the Field Team that they have permission to launch Seaglider.
- If any changes must be made to parameter values, do so in the cmdfile. If the pilot feels confident that the necessary changes have been made to the cmdfile, the pilot grants permission to the field team to launch. The updated cmdfile is transferred to Seaglider at the beginning of the launch sequence. Watch the file transfer in the ***comm.log*** file (tail –f comm.log).
- Were file transfers successful?
- Was a GPS fix obtained?
- Is the **$QUIT** directive at the end of the command file that was sent?

10. When the pilot gives permission to launch, and the field team types **Y**, Seaglider autonomous program takes over. It puts Seaglider in surface position (rolled to neutral, pitched fully forward, pumped to **$SM_CC**, typically maximum VBD for launch) and enters a normal surface phase: acquiring GPS1 and initiating a communication session via Iridium satellite telephone.

**FIGURE 6-7. Final OK to Launch**

```
Setting glider to pre-launch state ...
1681.594,HROLL,N,Roll commanded from -2.52 deg (2381) to 0.00 deg (2470)...
1682.670,HROLL,N,-0.2 deg (ad: 2464) MOTOR_DONE: ticks: 24 max 24v: 4.6mA avg 24v: 3.8mA minV 24v: 26.4V
1683.036,SMOTOR,N,GC TICKS/TIME: 27/28577
1683.090,HROLL,N,done.
1683.122,HROLL,N,Roll: No motion occurred.
1683.195,HPITCH,N,Pitch commanded from -6.68 cm (717) to -8.67 cm (80)...
1684.250,HPITCH,N,-7.1 cm (ad: 568) ..MOTOR_DONE: ticks: 15 max 24v: 4.6mA avg 24v: 3.8mA minV 24v: 26.3V
1686.183,SMOTOR,N,GC TICKS/TIME: 90/92125
1686.237,HPITCH,N,done.
1686.279,HPITCH,N,Pitch completed from -6.66 cm (722) to -8.64 cm (90) took 1.8 sec 78mA (286mA peak) 25.9Vmin 351 AD/sec 72 ticks
1686.459,HPITCH,N,TRACK: b: 92/0 a: 90/0 d: 2 o: 0
1686.543,HVBD,N,Bleed commanded from 581.84 cc (427) to 571.29 cc (470)...
1687.603,HVBD,N,579.1 cc (ad: 438 [448, 437]) ..MOTOR_DONE: ticks: 6 max 24v: 1224.9mA avg 24v: 209.4mA minV 24v: 26.0V
1689.095,SMOTOR,N,GC TICKS/TIME: 72/74284
1689.149,HVBD,N,done.
1689.191,HVBD,N,Bleed completed from 581.84 cc (427) to 571.29 cc (470 [479, 462]) took 1.6 sec 0mA (1081mA peak) 26.4Vmin 27 AD/sec 63 ticks
1689.382,HVBD,N,TRACK: b: 479/462 a: 478/462 d: -8 o: -8
1689.455,SUSR,N,query timeout lengthened, you have one hour to respond
Has the pilot given the OK to launch (no cancels launch, yes proceeds)? [N]
```

Step 10

11. **Both Pilot and Field Team**:
    - Immediately after the launch command is given to Seaglider, a communication session with the basestation starts.

- Both the field team and the pilot should watch to make sure that the session runs completely and properly. This is the last chance to abort the launch sequence easily before putting Seaglider in the water.

12. **Field Team:** If the communication session is acceptable, the field team should disconnect the communications cable at base of antenna and replace it with the water proof dummy plug.
13. **Field team:** Seaglider is now ready for deployment in the ocean. If not already on a ship at sea, load Seaglider and all ancillary equipment needed for the deployment on a boat. An example equipment checklist is in Appendix I, "Sample Field Kit Checklist" on page 439.
14. **Field Team:** Transit to the predetermined launch site. Keep the pilot informed of your progress and estimated time of arrival at the launch site.
15. **Field Team:** As the deployment site nears:
    a. Remove the protective covers from the science sensors.

       There are two caps on the conductivity cell, one covering the WET Labs sensor, two caps on the Sea-Bird dissolved oxygen sensor (pumped).

       **Note:** Not all sensors are installed on all gliders.

    b. Place a light tag line around Seaglider, just forward of the rudder so that the line rests in the grooves on the front edge.
    c. Place a safety line on the cradle and tie it off to the boat.

       This prevents its loss should someone let go of it during the deployment.
    d. Set up the acoustic deck unit, including putting a safety line on the transducer and tying it off to the boat and setting the correct transmit frequency for that Seaglider.

       The specific transmit and receive frequencies are listed in the Cal sheet of Seaglider's notebook.
16. **Field Team**: The field team notifies the pilot when they arrive at the final deployment site.
17. **Field Team**: Make sure that the tag line on Seaglider is tied off on the boat. With the boat captain's permission, deploy Seaglider and allow the fairing to fill with water.
18. **Field Team**: Notify the pilot that Seaglider has been deployed and report how Seaglider is sitting in the water.
    - How much of the rudder is showing?
    - How much of the antenna mast is visible.

      If Seaglider is riding well in the water, continue with the testing.

If Seaglider is riding very low in the water, and more than one-half of the antenna mast is under water there is an issue with the ballasting. Have the pilot verify that the external bladder is fully inflated. If it is not, this could be the issue. However, the question of why the external bladder is not fully inflated must be answered before proceeding. If the bladder is fully inflated (or nearly so), the ballasting weight needs to be verified and, if necessary, some lead attached to the outside of the pressure housing needs to be removed. The pilot determines if any lead has to be removed and from where.

19. **Field Team**:
    - With the captain's permission put the transducer of the acoustic recovery system over the side of the boat and down to a depth of 8-10 feet below the level of the boat's propeller.
    - Transmit a ping.

      Seaglider's transponder should respond. The range to Seaglider shows up on the screen of the acoustic deck unit.

      If Seaglider does not respond to the ping try several more times, slightly altering the depth of the transducer and the distance between Seaglider and the transducer.

      If the transponder still does not reply to the interrogation pings contact the pilot. The parameter **$XPDR_VALID** may have to be adjusted by the pilot with the new value sent to Seaglider via the cmdfile.

      If the transponder still does not respond to the transducer's interrogation there may be a problem with the acoustic locator system and Seaglider should be recovered.
20. **Pilot and Field Team**: If Seaglider is riding well in the water and the acoustic ranging system test went well, the pilot instructs the filed team to remove the tag line from Seaglider and standby while Seaglider makes its first dive(s).
21. **Pilot:** The pilot changes the directive in the cmdfile from **$QUIT** to **$RESUME** and the file is downloaded to Seaglider the next time it calls the basestation (**$T_RSLEEP** from the last time it called in). After Seaglider receives the new command file it starts the first (shallow) dive.
22. **Field Team:** Notify the pilot when Seaglider leaves the surface.
23. **Pilot:** Change the directive in the cmdfile from **$RESUME** to **$QUIT** so that when Seaglider completes its first dive it remains on the surface.

### ARGOS tag

Prior to deployment the SPOT5 tag will have to be switched into **Deployed mode**.

You can display and change the mode of the SPOT5 after exiting communications by using a magnet.

Swipe a magnet once across the center of the SPOT5's controller board. The LED will flash in a sequence that displays the mode of the SPOT5 tag.

- Standby mode LED pattern: **2 blinks, a 3 second pause, and 2 more blinks**
- Deployed mode LED pattern: **10 blinks**

Once the mode is displayed, the **LED will stay on for 2 seconds**.

- If you keep the magnet away from the SPOT5 while the LED is on, the SPOT5 will stay in its current mode.
- If you swipe a magnet across the board while the LED is on, you will toggle the mode. If the SPOT5 was toggled to the Standby mode, the LED will blink as described in "Standby mode" above. If the SPOT5 was toggled to the Deployed mode, the LED will blink as described in "Deployed mode" above.

In summary, a single swipe of the magnet displays the mode of the SPOT5. It takes two specifically timed swipes of the magnet to change the mode.

The Argos SPOT5 transmitter attaches to the GPS antenna, with two saddle brackets and attaching screws. Prior to installation, the SPOT5 operating settings and user parameters must be set.

For directions on Argos tag installation, operator settings and user parameters, see the iRobot, Argos SPOT5 Tag Setup and Installation "document# 4335290" or the Wildlife Computers, SPOT5 user manual.

CHAPTER 7

# *Pre-Launch Procedures*

This chapter describes testing of basic functions using the interactive self test and simulated dives. The following topics are covered:

- "Interactive Pre-launch Self Test" on page 188
- "Simulated Dives" on page 219

If your Seaglider has been disassembled since completing Chapter 3, "Setting Up the System" on page 33, reassemble all but the rudder and antenna assembly following the directions in "Assembling Seaglider" on page 41.

If your Seaglider is fully assembled, but moving it outside is easier without the antenna mast mounted to the aft end of Seaglider, detach it following the steps in "Detaching the Antenna Mast" on page 174.

**Note:** Do not disconnect the antenna cable for this procedure.

## *Interactive Pre-launch Self Test*

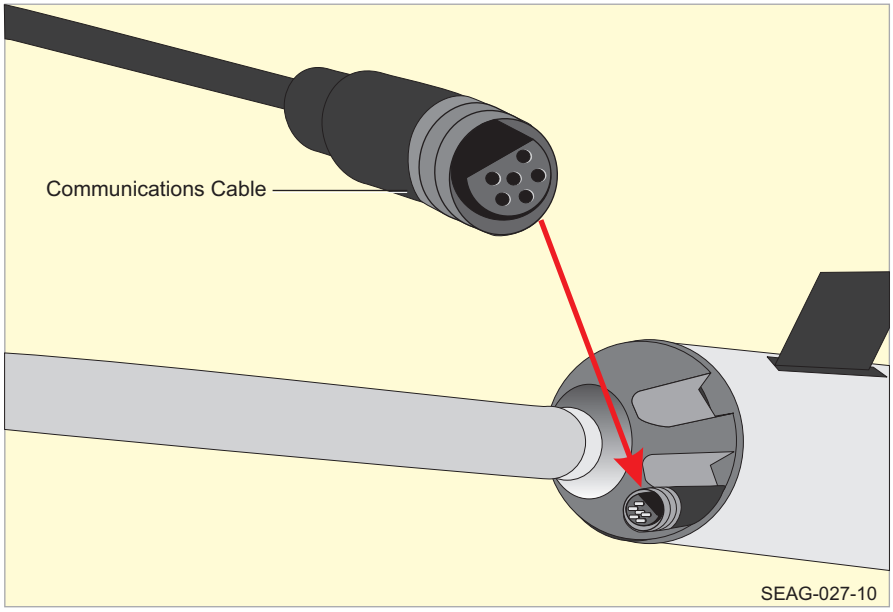
Perform a pre-launch self test after every time Seaglider is shipped or reassembled and before it is launched.

To conduct the interactive pre-launch self test:

1. Locate the calibration and configuration sheets in the notebook or on the CD that was shipped with Seaglider.
2. Move Seaglider outside.
3. Do one of the following:

| **If Seaglider is going to be...** | **Then...** |
|---|---|
| powered by an external power supply for these tests | connect the powered communications cable to Seaglider. |
| running on its internal battery packs for these tests | connect the 6-pin IE55 end of the non-powered communications cable to the communications port located at the base of Seaglider's antenna mast. |

**FIGURE 7-1. Communications Cable Attached to Seaglider**

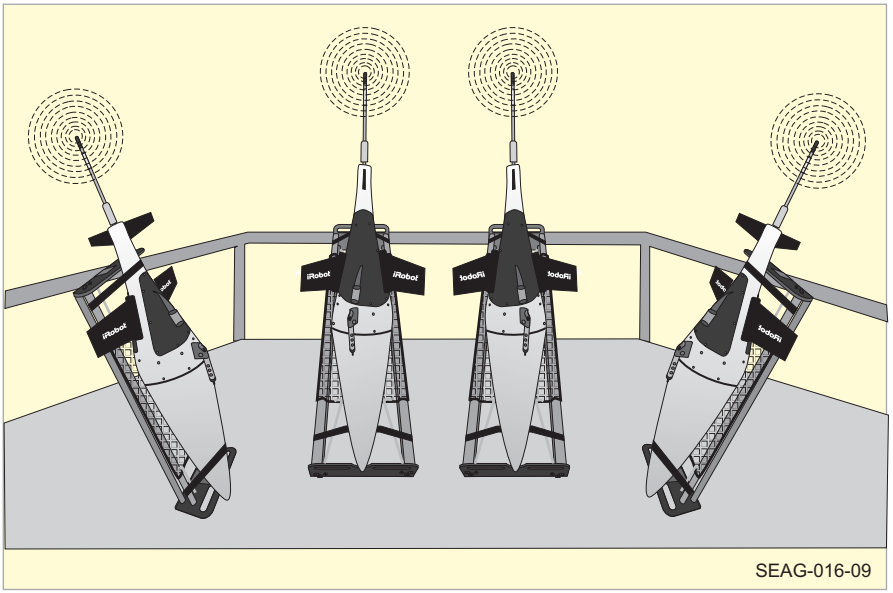



**Caution:** Be sure to line up the connector in the proper orientation. Slide the connector in slighty to ensure that the pins are aligned and the connector is properly seated before tightening the knurled metal ring.

4. Position Seaglider, in its cradle, nose down, tail up and make sure the antenna is within 40° of vertical and has an unobstructed view of the sky. (Figure 7-2)

   Make sure that Seaglider is secure in this position. Tying it to the secured fixture it is leaning against is recommended.

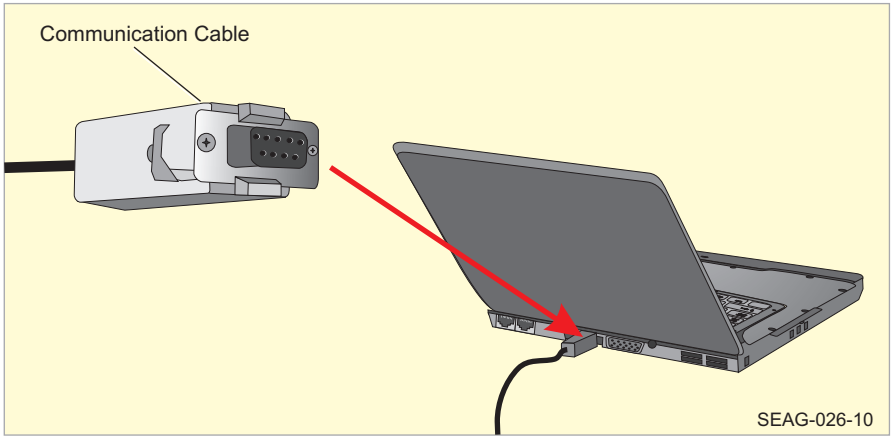
**FIGURE 7-2. Seaglider Positioned for Self Test**




5. If Seaglider is being powered externally, set up the dual voltage power supply, checking both voltage output and current limit. Set one supply to 24V with a 2 amp current limit and the other supply to 10V with a 2 amp current limit. Connect the banana plugs on the powered communications cable to the power supplies.

6. Connect the communication cable to the DB9 connector on the laptop and power up the laptop if it has not already been done.

**FIGURE 7-3. Connecting the Communication Cable to the Laptop**




7. Start the terminal emulation program and start a screen capture file or a session log-file on the laptop.
8. Turn on the power supplies, if Seaglider is being powered externally.
9. Wand on Seaglider. See page 50 for instructions on wanding on Seaglider.
10. When the output starts scrolling on the computer screen, press ENTER within one minute.

    See Figure 7-4.

    **Note:** While performing the following steps, watch the process on the laptop and note if there are any errors or warnings. The screen capture file can also be checked at the end of the test for warnings and errors.

    **Note:** Pressing ENTER after any query results in the default answer/value, shown in brackets,[ ], after the question, being used.

**FIGURE 7-4. Seaglider Start-Up on the Laptop]**

```
Version 66.07/18:775 starting up Mar 29 2011 10:36:24
Seaglider operating software developed and maintained by
Applied Physics Laboratory, University of Washington
Copyright 2003-2010, University of Washington
Built with SCENARIO, without ICE, without RAFOS, without KERMIT,
      without ARS, without MMODEM, without PAAM, without AQUADOPP, without SAILCT,
     without DEEPGLIDER, without HEAP RECYCLE, with REV_C, without LUA
with serial and logger device sensor integration facilities
developed by Integrative Observational Platforms Group, APL-UW
_mtop = 0x2fefff, _mbot = 0x28cdb8, _mcur = 0x28e000
glmalloc_init: glmalloc_data_init = 0x0
glmalloc_init: avail_bytes = 434175, requested_bytes = 358400
glmalloc_init: Allocated 358400 bytes.
1301395235.048,SSENSOR,N,reading cnf GPCTDDO.cnf
1301395235.409,SSENSOR,N,assigned WL_BBFL2VMT to sensor slot 1 (p = 83)
1301395235.594,SSENSOR,N,assigned GPCTD to logger sensor slot 1 (p = 85)
1301395235.750,SSENSOR,N,assigned empty to logger sensor slot 3 (p = 0)
1301395237.358,SPOWER,N,Iridium_during_init saved current=0.10300
1301395237.539,SPOWER,N,Iridium_during_connect saved current=0.16000
1301395237.725,SPOWER,N,Iridium_during_xfer saved current=0.22300
1301395237.907,SPOWER,N,GPS saved current=0.02696
1301395238.067,SPOWER,N,TT8 saved current=0.01496
1301395238.227,SPOWER,N,TT8_Active saved current=0.01421
1301395238.397,SPOWER,N,TT8_Sampling saved current=0.03743
1301395238.570,SPOWER,N,TT8_CF8 saved current=0.04717
1301395238.737,SPOWER,N,TT8_Kalman saved current=0.05915
1301395238.908,SPOWER,N,Analog_circuits saved current=0.01200
1301395239.085,SPOWER,N,Compass saved current=0.01573
1301395239.242,HCOMPASS,N,Active compass is SP3003
1301395239.413,SSYS,N,Changing capture file
1301395239.518,SSENSOR,N,sensor SBE_CT not found in installed sensors!
1301395239.655,SSENSOR,N,sensor SailCTx not found in installed sensors!
1301395239.789,SSENSOR,N,no CT sensor configured
1301395244.621,SUSR,N,Glider 524, mission 6, dive 5, booted Tue Mar 29 10:40:44 2011, boot #374
1301395245.510,HXPDR,N,max range = 60.000000 m, timeout = 80 ms
1301395245.777,SSURF,N,Hit CR within 1 minute to start, otherwise will go to recovery ...
Set current date and time
<CR> for default [03/29/2011 10:40:48]
Are you running on external (bench) power? [N] y
1301395252.545,SPOWER,N,Not tracking power consumption!

------ Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 2
```

**11.** Set the current date and time (GMT) in response to the query (Figure 7-4).

A time within 12 hours is acceptable because Seaglider gets an accurate time from its first GPS fix and resets the internal clock. Note that the format must be in mm/dd/yyyy hh:mm:ss, with no missing values or extra spaces.

**12.** When prompted “Are you running on external (bench) power?,” (Figure 7-4), do one of the following:

- Type **Y** (for Yes) to indicate that you are using a powered communications cable, and press ENTER.
- Press ENTER to accept the default (N for No) if you do not have an external power source (using non-powered communications cable).

Once the ‘Are you running on external (bench) power question is answered, the main Seaglider menu appears (Figure 7-4).

(If your Seaglider is equipped with a GPCTD then continue with step 13, otherwise skip to step 14)

**13.** Turn off the GPCTD.

**a.** On units equipped with GPCTD it is necessary to turn off the GPCTD pump during self-tests and simulated dives. The pump may be damaged by running dry so this is necessary to comply with manufacturer's recommended operating procedures.

**Note:** Do not skip any steps. Failure to perform all steps as indicated can result in improper operation and/or tracking of power consumption of the GPCTD.

**FIGURE 7-5. Loggers Menu**

```
|----- Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 2

------ Hardware Menu ------
*Motors and VBD
     1 [pitch   ]  Pitch control
     2 [roll    ]  Roll control
     3 [vbd     ]  VBD control
*Fixed Devices
     4 [super   ]  Supervisor
     5 [pressure]  Pressure sensor
     6 [compass ]  Compass (tcm2)
     7 [gps     ]  GPS
     8 [modem   ]  Modem (xmodem mode)
     9 [intpress]  Internal pressure
    10 [altim   ]  Altimeter
    11 [sensors ]  Sensors
    12 [loggers ]  Loggers
*Other
    13 [batt    ]  Batteries and fuel gauges
    14 [lowlevel]  Low-level hardware (IO,A-D,CF)
    15 [misc    ]  Miscellaneous (travel, timeouts, date/time)
    16 [develop ]  Developer tests
    CR)  Return to previous
Enter selection (1-16,CR): 12

------ Loggers menu ------
     1 [pc      ]  GPCTD
     2 [        ]  empty
    CR)  Return to previous
Enter selection (1-2,CR): 1
```

**b.** Press **2** and press ENTER to enter the Hardware Menu

**c.** Press **12** and press ENTER to enter the Loggers Menu

**d.** Press **1** and press ENTER to enter the GPCTD menu

**FIGURE 7-6. Turning off the GPCTD**

```
├----- Logger device test menu ------
     1 [on       ]  Turn on
     2 [off      ]  Turn off
     3 [selftest]  Selftest
     4 [sample   ]  Report a sample
     5 [syncclk ]  Synchronize device clock to TT8
     6 [clock    ]  Read device clock
     7 [file     ]  Get file from device
     8 [action   ]  Execute logger action
     9 [config   ]  Show configuration
    10 [edit     ]  Edit configuration
    11 [direct   ]  Direct comms
    CR)  Return to previous
Enter selection (1-11,CR): 11
35.883,HGPCTD,N,RX = 10, TX = 11
36.579,SUSR,N,Starting breakable loop - Ctrl-Q to end

S>GetCD
<ConfigurationData DeviceType = 'SBE Glider Payload CTD' SerialNumber = '70100016'>
   <SampleDataFormat>HEX engineering</SampleDataFormat>
   <RS232ForceOn>yes</RS232ForceOn>
   <TxRealTime>no</TxRealTime>
   <SampleInterval>1</SampleInterval>
   <SampleMode>Continuous</SampleMode>
   <AutoRun>no</AutoRun>
   <SBE43>yes</SBE43>
   <MinCondFreq>3212.0</MinCondFreq>
   <CustomPumpMode>no</CustomPumpMode>
</ConfigurationData>
S>minCondfreq=10000
S>GetCD
<ConfigurationData DeviceType = 'SBE Glider Payload CTD' SerialNumber = '70100016'>
   <SampleDataFormat>HEX engineering</SampleDataFormat>
   <RS232ForceOn>yes</RS232ForceOn>
   <TxRealTime>no</TxRealTime>
   <SampleInterval>1</SampleInterval>
   <SampleMode>Continuous</SampleMode>
   <AutoRun>no</AutoRun>
   <SBE43>yes</SBE43>
   <MinCondFreq>10000</MinCondFreq>
   <CustomPumpMode>no</CustomPumpMode>
</ConfigurationData>
S>start
<!--start logging-->
S>    0.08, 15.4768, 0.00004
    0.09, 15.4771, 0.00004
    0.08, 15.4770, 0.00003
    0.07, 15.4778, 0.00003
    0.08, 15.4782, 0.00003
    0.08, 15.4788, 0.00003
    0.09, 15.4789, 0.00003
    0.09, 15.4798, 0.00003
    0.07, 15.4797, 0.00003
    0.08, 15.4799, 0.00003
    0.08, 15.4808, 0.00003
    0.08, 15.4811, 0.00003

<logging stopped/>
S>
------ Logger device test menu ------
     1 [on       ]  Turn on
     2 [off      ]  Turn off
     3 [selftest]  Selftest
     4 [sample   ]  Report a sample
     5 [syncclk ]  Synchronize device clock to TT8
     6 [clock    ]  Read device clock
     7 [file     ]  Get file from device
     8 [action   ]  Execute logger action
     9 [config   ]  Show configuration
    10 [edit     ]  Edit configuration
    11 [direct   ]  Direct comms
    CR)  Return to previous
Enter selection (1-11,CR): 2
```

**e.** Press **11** and press ENTER to go into direct serial communications with the GPCTD

**f.** Press ENTER until a 'S>' prompt appears

**g.** Type **GetCD** and press ENTER

- Find the value labeled MinCondFreq and record that value so you can use it later to turn the pump back on.

**h.** Type **MinCondFreq=10000** and press ENTER

**i.** Type **GetCD** and press ENTER

- Find the value labeled MinCondFreq and confirm it has been set to 10000

j. Type **Start** and press ENTER

k. Wait for 30 secs. (You may be able to hear the pump is stopped, though the sound is hard to hear externally.)

l. Type **Stop** and press ENTER to stop system and get prompt back. You will not see the word "Stop" displayed – but a message will appear that the "logging stopped."

m. Press Ctrl-Q to exit direct communications with the GPCTD

n. Once back at the GPCTD menu, type **2** and press enter to turn off the GPCTD

o. Press ENTER until you return to the main menu

p. Wand the glider off.

q. Wand the glider back on to continue testing/operations.

14. Test Seaglider's altimeter/transponder:

**FIGURE 7-7. Altimeter Check**

```
├----- Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 2

------ Hardware Menu ------
*Motors and VBD
     1 [pitch   ]  Pitch control
     2 [roll    ]  Roll control
     3 [vbd     ]  VBD control
*Fixed Devices
     4 [super   ]  Supervisor
     5 [pressure]  Pressure sensor
     6 [compass ]  Compass (tcm2)
     7 [gps     ]  GPS
     8 [modem   ]  Modem (xmodem mode)
     9 [intpress]  Internal pressure
    10 [altim   ]  Altimeter
    11 [sensors ]  Sensors
    12 [loggers ]  Loggers
*Other
    13 [batt    ]  Batteries and fuel gauges
    14 [lowlevel]  Low-level hardware (IO,A-D,CF)
    15 [misc    ]  Miscellaneous (travel, timeouts, date/time)
    16 [develop ]  Developer tests
    CR)  Return to previous
Enter selection (1-16,CR): 10

------ Altimeter/XPDR Menu ------
     1 [ping    ]  Ping the altimeter
     2 [config  ]  Upload configuration to altimeter
     3 [count   ]  Query the transponder ping count
     4 [xpdrcfg ]  Update XPDR configuration from parameters
     5 [direct  ]  Direct comms with altimeter
     6 [param   ]  Edit altimetry parameters
    CR)  Return to previous
Enter selection (1-6,CR): 3
956.913,HXPDR,N,ping count 0

------ Altimeter/XPDR Menu ------
     1 [ping    ]  Ping the altimeter
     2 [config  ]  Upload configuration to altimeter
     3 [count   ]  Query the transponder ping count
     4 [xpdrcfg ]  Update XPDR configuration from parameters
     5 [direct  ]  Direct comms with altimeter
     6 [param   ]  Edit altimetry parameters
    CR)  Return to previous
Enter selection (1-6,CR): 3
1033.230,HXPDR,N,ping count 3
```

**a.** Set up the acoustic deck box.

**b.** Set the correct transmit frequency on the deck box. This frequency is in the notebook that was shipped with Seaglider.

**c.** Put the transducer next to the nose of Seaglider, touching the fairing.

**d.** From the Main Menu, type **2** (Hardware tests and monitoring) and press ENTER. (Figure 7-7)

**e.** Select **10**: Altimeter and press ENTER. (Figure 7-7)

**f.** Select **3**: Query the transponder ping count. Ping count should be 0. If not, select 3 again until ping count is 0.

**g.** Send a ping from the deck box to Seaglider. The system displays a response on the screen of the deck box and on the laptop. The display on the deck box is automatic. To see the ping count on the laptop (Figure 7-7), select 3: Query the transponder ping count and the count should go up to the number of times the deck box was pinged.

Seaglider returns to the Altimeter/XPDR menu.

h. If there is no response to the ping, do the following:
   - Send another ping.
   - If there is still no response, move the transducer around the nose a bit.
   - If there still is no response after several more ping attempts, there may be an issue with the acoustic locator system or there could just be an air gap between the two transducers. You should investigate and resolve the problem, however, the acoustic locator system is not critical to the rest of the self test so you may choose to continue the testing and investigate the acoustic system problem separately.

15. Press ENTER until the Main Menu appears (Figure 7-8).
16. Type **5** for Pre-launch options, and then press ENTER (Figure 7-8). The Launch Menu appears.

**FIGURE 7-8. Interactive Self-Test**

```
------ Main Menu ------
     1 [param    ]  Parameters and configuration
     2 [hw       ]  Hardware tests and monitoring
     3 [modes    ]  Test operation modes and files
     4 [pdos     ]  PicoDOS commands (and exit)
     5 [launch   ]  Pre-launch
Enter selection (1-5,CR): 5

------ Launch Menu ------
     1 [scene    ]  Set scenario mode
     2 [selftest]  Perform interactive self test
     3 [autotest]  Perform autonomous self test
     4 [uploadst]  Upload self-test results
     5 [reset    ]  Reset dive/run number
     6 [test     ]  Test Launch!
     7 [sea      ]  Sea Launch!
    CR)  Return to previous
Enter selection (1-7,CR): 2
```

17. Type **2** (Perform interactive self test), and then press ENTER. (Figure 7-8)
18. The prompt “Remove old data/log files (for self-test before sea launch)? [N]” appears.

**Warning:** Selecting ‘Yes’ removes all data files still on Seaglider from prior missions and tests. Be sure that those files are saved elsewhere BEFORE selecting ‘Yes’. If it is unknown whether the files have been saved, select N.

Figure 7-9 shows file removal.

**FIGURE 7-9. Removing Old Data/Log Files**

```
Remove old data/log files (for self-test before sea launch)? [N]  y
4.171,SUSR,N,Removing old data and log files
Deleting SG0000KZ.A...
Deleting SG0000LZ.A...
Deleting SG0001KZ.A...
Deleting SG0001DZ.A...
Deleting SG0001LZ.A...
5 files deleted.
4.047,SUSR,N,Tue Mar 29 10:41:06 2011
```

**19.** At the prompt "Can the antenna be used for GPS and communications? [Y]," press ENTER to accept the default answer, Yes (Figure 7-10).

**FIGURE 7-10. Seaglider ID Verification and Communications Set Up**

```
Can the antenna be used for GPS and communications? [Y]
Verify GPS and communications parameters? [Y]
$ID
New value? [524.00]
$MISSION
New value? [6.00]
20.765,SUSR,N,Current password is 524680
New password (15 char max length, CR to leave unchanged):
23.299,SUSR,N,Current telnum is 19195726671
New telnum (15 char max length, CR to leave unchanged):
25.834,SUSR,N,Current altnum is 19195726670
New altnum (15 char max length, CR to leave unchanged):
27.705,SUSR,N,$CALL_TRIES: 5 OK
27.808,SUSR,N,$N_NOCOMM: 1 OK
$T_GPS
New value? [15.00]
29.898,SUSR,N,Place glider in position for communications..
Hit CR to continue
```

**Caution:** This answer is critical. The answer Yes means that Seaglider's antenna is in position for communications and ready to be launched.

**20.** At the prompt "Verify the GPS and communications parameters? [Y]," press ENTER to accept the default answer, Yes (Figure 7-10).

**21.** Verify Seaglider's ID number by comparing the ID on Seaglider's aft fairing, just forward of where the antenna mast is bolted into the fairing, with the number on the screen (Figure 7-10).

**22.** The ID is a three digit number.

**23.** If the ID numbers agree, press ENTER to accept the default (Yes) (Figure 7-10).

**24.** If the IDs do not agree, investigate and correct the cause of the discrepancy before continuing with the testing. The calibration coefficients are assigned to a Seaglider based on the ID and Seaglider's serial number is based on the ID.

**25.** Verify the Mission Number, and then press ENTER to accept the default, 0, for the first self test (Figure 7-10).

Subsequent self tests are numbered sequentially by Seaglider. You can override the Seaglider-generated number with your own number.

You define subsequent mission numbers either here of through the cmdfile from the basestation.

The current password appears.

**26.** Verify the current password (Figure 7-10).

At the prompt "New password (15 char max length, CR to leave unchanged)," type a new password and press ENTER, or press ENTER to leave the password unchanged (Figure 7-10).

The password installed at the factory is six digits:

- The first three digits are Seaglider's ID number.
- The last three digits are 680, if Seaglider's ID is even, or 791, if Seaglider's ID is odd.

If you want to use a different password, it cannot be more than 15 characters. The characters can be any alphanumeric string, but no punctuation or special characters are allowed.

If you change Seaglider's password installed at the factory, you must also make a corresponding change on the basestation for this Seaglider. Changing the password is NOT recommended.

If the password is incorrect, the test failed.

**Caution:** Investigate and correct the cause of the discrepancy before continuing with the testing. A Seaglider cannot call the basestation unless the password is the same on both the basestation and Seaglider.

**27.** Set the primary telephone number.

At the prompt "New telnum (15 char max length, CR to leave unchanged)," verify that the number displayed matches the number of the telephone line going into the first basestation modem (Figure 7-10).

The telephone number is unique to that phone line. It starts with the international country code, without leading zeros (for example, "1" for the US), then city/area code and number. There are no spaces or other interrupting characters between the country code, city/area code or number.

Do one of the following:

- If the telephone numbers match, press ENTER to accept without making changes (Figure 7-10).
- If the telephone numbers do not match, enter the correct number and press ENTER (Figure 7-10).

**28.** Set the alternate telephone number.

At the prompt "New altnum (15 char max length, CR to leave unchanged," verify that the value displayed matches the number of the telephone line going into the second basestation modem (Figure 7-10).

The alternate number is unique to that phone line.

Do one of the following:

- If the telephone numbers match, press ENTER to accept without making changes (Figure 7-10).
- If the telephone numbers do not match, enter the correct number and press ENTER (Figure 7-10).

**29.** Put Seaglider in position for communications, if it is not already there. Seaglider should be outdoors, nose down with the antenna pointed to within 40 degrees of vertical.

**30.** When Seaglider is in position, press ENTER (Figure 7-10).

**31.** At the prompt "Allow the GPS to acquire the almanac? [N]," should press ENTER to accept the default 'No.'

**FIGURE 7-11. Checking GPS and Iridium**

```
32.064,SUSR,N,---- Checking GPS ----
32.149,HGPS,N,Obtaining GPS fix
53.340,HGPS,N,0 recent almanac sentences returned, consider acquiring new almanac data
53.456,HTT8,N,Updating parameter $T_GPS_CHARGE to -6158.4648
Allow the GPS to acquire the almanac? [N]
115.910,HGPS,N,Acquiring GPS fix ...
117.756,HGPS,N,VVVVVVVVVVVVA
1497732.001,HGPS,N,sync sentence $GPRMC,184313,A,3556.3498,N,07854.1015,W,000.0,000.0,150411,008.7,W*7D
1497732.131,HGPS,N,set 2011/04/15 18:43:14
1497732.572,HGPS,N,confirm $GPRMC,184314,A,3556.3499,N,07854.1015,W,000.0,000.0,150411,008.7,W*7B
1497732.718,HGPS,N,150411, 184312, 3556.349365, -7854.101562 1.700000 16/16 seconds
1497733.333,HTT8,N,Updating parameter $T_GPS_CHARGE to -6159.0952
1497735.786,SUSR,N,---- Checking Iridium phone ----
1497737.068,HPHONE,D,Copyright(C) 2009
1497737.162,HPHONE,D,NAL Research Corporation
1497737.269,HPHONE,D,Iridium Modem Model: A3LA-XG
1497737.334,HPHONE,N,error collecting signal strength from phone
1497738.417,HPHONE,D,lat: 0.000000, lon: 0.000000, time: Thu Jan  1 00:00:00 1970
1497740.554,HPHONE,D,AT+CREG?
1497740.620,HPHONE,D,+CREG:000,004
1497740.669,HPHONE,N,WARNING: Phone reported 4 stat on CREG?
1497740.748,HPHONE,D,
1497740.780,HPHONE,N,CREG failed - phone not registered
Continue self-test? [N] y
```

**Note:** Seaglider has 15 minutes (T_GPS) to acquire a GPS position. If it does not, once the test is started, the test fails.

Seaglider then tests the Iridium phone connection to the satellites. Often this fails because the code does not wait long enough to measure signal strength or register on the Iridium system. At this time, you can ignore the failed 'CREG' because the self test tries to make another Iridium call at the end of the self test procedure.

32. At the prompt “Continue to Self Test?[N],” type **Y** (for Yes), and then press ENTER (Figure 7-11).
33. Seaglider checks for bathymetry maps (Figure 7-12).
34. If the self test returns the message “Warning: No bathymetry file covers the current locations! Continue self test?[N],” do one of the following:
    - If the test is being conducted on land, where there are no bathymetry files available, or if on the water, but it is known that no bathymetry maps are loaded for the present area, type **Y** (for Yes), and press ENTER (Figure 7-12).
    - If bathymetry maps are supposed to have been loaded, type N (default), and press ENTER to not continue the self test. The test fails (Figure 7-12). Determine the cause for Seaglider’s inability to find the map(s). Check the file names. The format must be bathymap.nnn where nnn is the map’s number.

**FIGURE 7-12. Checking Bathymetry Data**

```
1497750.851,SUSR,N,---- Checking bathymetry data ----
1497750.964,SUSR,N,Normal Heap: 0+73727 bytes
1497751.103,SGLMALLOC,N,glheap_walk: 354980 bytes free, 2 blocks free, 2820 bytes alloc, 23 blocks alloc
1497755.913,SBATHY,N,Loaded bathymap.006 [156 100] 100m (LL) 47.6800,-122.5000 (UR) 47.8204,-122.3663
1497758.465,SBATHY,N,Loaded bathymap.007 [ 56  75] 100m (LL) 47.6500,-122.5000 (UR) 47.7004,-122.3998
1497769.116,SBATHY,N,Loaded bathymap.014 [125 147] 500m (LL) 28.4643,-89.0336 (UR) 29.0268,-88.2812
1497769.464,SBATHY,N,WARNING: No bathymetry file covers the current location (3556.349365,-7854.101562)!
1497770.524,SGLMALLOC,N,glheap_walk: 318024 bytes free, 3 blocks free, 39704 bytes alloc, 25 blocks alloc
Continue self-test? [N] y
```

35. After the bathymetry check, compass and calibration file checks are done.

    Verify that the values for pitch, roll, hard iron and soft iron are non-zero. If the values are zero, the test fails (Figure 7-13).

**FIGURE 7-13. Compass Calibration**

```
1497776.543,SUSR,N,---- Checking compass and calibration files ----
1497776.746,HCOMPASS,N,Compass 1, SP3003
1497776.991,HCOMPASS,N,pitch A,B,C,D: -0.004800 1.012200 -0.008200 -0.002100
1497777.293,HCOMPASS,N,roll A,B,C,D:  -0.006700 -7.096700 8.482200 0.001000
1497777.586,HCOMPASS,N,hard iron P,Q,R: 22.632999 51.560799 36.951199
1497777.834,HCOMPASS,N,soft iron: 1.002400 -0.015200 -0.004200
1497778.069,HCOMPASS,N,           0.001000 1.062200 0.002400
1497778.307,HCOMPASS,N,           0.011000 0.017500 1.060800
1497779.828,HCOMPASS,N,compass filter disabled ok
1497780.659,HCOMPASS,N,1, SP3003, Hdg: 20.00 deg Rol: 134.96 deg Pit: -65.32 deg Tmp: 24.70 C
```

**36.** Verify that the critical operational parameters are ok (Figure 7-14).

**FIGURE 7-14. Critical Operational Parameters**

```
1497780.843,SUSR,N,---- Verifying critical operational parameters ----
1497781.002,SUSR,N,$RELAUNCH: 1 OK
1497781.131,SUSR,N,$AD7714Ch0Gain: 128 OK
1497781.226,SUSR,N,$SIM_PITCH: 0 OK
```

**37.** The glider will now report a full listing of all the installed hardware and software (Figure 7-15).

**FIGURE 7-15. Reporting Hardware Configuration and Software Versions**

```
1497781.315,SUSR,N,---- Reporting hardware configuration ----
1497781.456,SUSR,N,Sensor in slot 1 is WL_BBFL2VMT on port 3, TPU06/TPU07, nominally 'Optics 1'
1497781.625,SUSR,N,Sensor in slot 2 is not installed
1497781.736,SUSR,N,Sensor in slot 3 is not installed
1497781.847,SUSR,N,Sensor in slot 4 is not installed
1497781.958,SUSR,N,Sensor in slot 5 is not installed
1497782.069,SUSR,N,Sensor in slot 6 is not installed
1497782.206,SUSR,N,Logger sensor in logger slot 1 is GPCTD on port 5, TPU10/TPU11 (mux channel 0), nominally 'Optics 2'
1497782.410,SUSR,N,Logger Sensor in logger slot 2 is not installed
1497782.561,SUSR,N,Logger sensor in logger slot 3 is empty on port 0, TPU-1/TPU-1, nominally 'null'
1497782.739,SUSR,N,Logger Sensor in logger slot 4 is not installed
1497782.865,SUSR,N,Motherboard is Rev.B.1
1497782.957,SUSR,N,Phone is Iridium9522
1497783.045,SUSR,N,GPS is Garmin_GPS15H
1497783.134,SUSR,N,Compass is SP3003
1497783.220,SUSR,N,Spare compass is not installed
1497783.322,SUSR,N,XPDR is AAE_955
1497783.466,SUSR,N,External pressure sensor gain (128.000000) consistent with Paine
1497784.468,SUSR,N,---- Reporting software version ----
1497784.613,SUSR,N,Version: 66.07
1497784.697,SUSR,N,Compiled on: Mar 29 2011 10:36:24
1497784.804,SUSR,N,From SVN version 18:775
1497784.896,SUSR,N,Compiler: gcc
1497784.978,SUSR,N,Compiled with SCENARIO support
1497785.083,SUSR,N,Compiled without ICE support
1497785.185,SUSR,N,Compiled without RAFOS support
1497785.291,SUSR,N,Compiled without Kermit support
1497785.397,SUSR,N,Compiled without ARS support
1497785.499,SUSR,N,Compiled without PAAM support
1497785.603,SUSR,N,Compiled without AQUADOPP support
1497785.713,SUSR,N,Compiled without SAILCT support
1497785.820,SUSR,N,Compiled without MicroModem support
1497785.933,SUSR,N,Compiled without DeepGlider support
1497786.046,SUSR,N,Compiled without Heap Recycle support
1497786.164,SUSR,N,Compiled with Rev C Motherboard support
1497786.281,SUSR,N,Compiled without LUA support
1497786.393,SUSR,N,NVRAM: 178 parameters, 716 bytes, 186 bytes in utility storage
1497786.538,SUSR,N,Buffers: static
1497786.619,SUSR,N,Active (2Mhz)
1497786.700,SUSR,N,PicoDOS version: C82#2651-1.63b
1497786.804,SUSR,N,Launched as: MAIN
1497786.893,SUSR,N,Normal Heap: 0+73727 bytes
1497787.036,SGLMALLOC,N,glheap_walk: 317984 bytes free, 3 blocks free, 39720 bytes alloc, 26 blocks alloc
1497787.212,SUSR,N,Sensors: WL_BBFL2VMT
1497787.860,SUSR,N,MAIN.RUN: 490112 bytes
```

**38.** The glider will next report a full listing of all the directory contents (Figure 7-16).

**FIGURE 7-16. Reporting Directory Contents]**

```
18158.855,SUSR,N,---- Reporting directory contents ----
BATHYMAP.001          33792
BATHYMAP.002          38272
BATHYMAP.003          26752
BATHYMAP.004          48896
BATTERY                 494
BG072110                482
CMDFILE                 709
RUN099.CFG              322
SCIENCE                 129
SG001PRM.TXT           2785
BG012611                484
TARGETS                 361
MAIN.RUN             490112
GPCTD.CNF               768
TCM2MAT.001             192
TCM2MAT.526             256
TCM52600.AVE           4830
TCM52600.RAW          85037
THISDIVE.DAT           3775
THISDIVE.KAP           4118
THISDIVE.LOG           5791
THISDIVE.WRK            129
6606.RUN             469720
CURRENTS                256
LOGGERS.CMD             109
SG0000KZ.A             2710
BG010311                497
OLDMAIN.RUN          489988
SG0000LZ.A             1374
SG0001KZ.X02           2137
SG0001KZ.A            10329
SG0001DZ.A             1209
SG0001LZ.A             2619
SG526PRM.TXT           2899
RECEIVE.000             256
RECEIVE.001               0
RECEIVE.002             768
QSP2150.CNF             128
RECEIVE.003             256
RECEIVE.004             384
WLBB2F.CNF              256
WLBB2FL.CNF             256
WLBBFL2.CNF             256
660701.RUN           490240
BG020711                488
2225621 bytes used; 45 files
```

**39.** Verify the pressure sensor.

**FIGURE 7-17. Checking Pressure Sensor**

```
1497795.211,SUSR,N,---- Checking pressure sensor ----
Are you at sealevel? [Y]
Number of samples to collect and average: [11]
Show all samples? [N]
1497802.425,HPRES,N,                A/D         Volts     Meters
1497802.671,HPRES,N, Mean:          415373.6    0.482591   0.01
1497802.937,HPRES,N, RMS:           415373.8    0.482592   0.01
1497803.148,HPRES,N, Min:           414718      0.481830   -0.04
1497803.361,HPRES,N, Max:           416149      0.483492   0.07
1497803.538,HPRES,N, Peak-to-peak:  1431        1.662571   uV

1497803.705,HPRES,N, Volts per bit: 1.161825e-09

1497803.811,HPRES,N,Y-intercept is set correctly
```

**a.** At the prompt “Are you at sea level? [Y],” type **Y** (for Yes, default) if you are at sea level or N (for No), and then press ENTER (Figure 7-17).

**b.** At the prompt “Number of samples to collect and average: [11],” press ENTER to accept the default of 11 (Figure 7-17).

**c.** At the prompt “Show all samples? [N],” press ENTER to accept the default N (Figure 7-17).

The results from the sea level calibration display.

**d.** At the prompt "If truly at sea level, this data suggests it should be xx.x(psig). Accept new value?[N], do one of the following:

- If Seaglider is at sea level and you want to accept the new value, type **Y** (for Yes), and then press ENTER.
- If Seaglider is not at sea level, press ENTER to accept the prior value.

(If your Seaglider is equipped with a GPCTD then continue with step 40, otherwise skip to step 41.)

**40.** The GPCTD configuration is now verified. Values for "clock sync string is" and "fmt into strftime ="are read from the configuration file loaded for the GPCTD sensor. The raw data for the sensor is returned in the form of the hex string at the end of the verification. (Figure 7-18)

**FIGURE 7-18. Checking GPCTD**

```
1497803.925,SUSR,N,---- Checking GPCTD ----
1497804.018,HGPCTD,N,syncing clock
clock sync string is %r%[S>]RESETLOGGING%r%[confirm%n]RESETLOGGING%r%[S>]%{DATETIME=%m%d%Y%H%M%S}%[S>]
fmt into strftime = [DATETIME=%m%d%Y%H%M%S]
1497810.688,HGPCTD,N,sending [DATETIME=04152011184435]sleeping 92176 ticks, sync will be 0x0d
1497816.247,HGPCTD,N,sample: TS
003CD571540138C0BADD
```

**41.** Check the pitch motor and values.

**FIGURE 7-19. Checking Pitch Motor]**

```
S>---- Checking pitch, roll, and VBD. ----
Hit ctl-Q to interrupt current test and continue...
1497817.346,SUSR,N,---- Checking pitch motor ----
Verify pitch SW limits and timeouts? [Y]
$PITCH_MIN
New value? [80.00]
$PITCH_MAX
New value? [3853.00]
$PITCH_TIMEOUT
New value? [16.00]
1497845.650,SUSR,N,Pitching up...
1497845.851,HPITCH,N,Pitch commanded from -3.63 cm (689) to 6.26 cm (3853)...
1497847.206,HPITCH,N,-3.4 cm (ad: 778) ...................
1497857.981,HPITCH,N,3.4 cm (ad: 2933) ........Pitch motor timeout!
.1497863.102,SMOTOR,N,MOTOR_DONE: ticks: 5 max 24v: 3.1mA avg 24v: 2.3mA minV 24v: 23.7V
1497863.259,SMOTOR,N,GC TICKS/TIME: 648/650661
1497863.354,HPITCH,N,done.
1497863.500,HPITCH,C,Pitch completed from -3.63 cm (689) to 6.20 cm (3835) took 16.0 sec 252mA (822mA peak) 23.5Vmin 196 AD/sec 640 ticks; 1 errors
1497863.934,HPITCH,N,TRACK: b: 3824/0 a: 3824/0 d: 16 o: 11
1497864.050,SUSR,N,Pitching down...
1497864.254,HPITCH,N,Pitch commanded from 6.20 cm (3835) to -5.53 cm (80)...
1497865.629,HPITCH,N,5.8 cm (ad: 3701) ...................
1497876.379,HPITCH,N,-5.2 cm (ad: 182) .MOTOR_DONE: ticks: 17 max 24v: 3.1mA avg 24v: 2.3mA minV 24v: 23.8V
1497877.423,SMOTOR,N,GC TICKS/TIME: 480/482298
1497877.519,HPITCH,N,done.
1497877.665,HPITCH,N,Pitch completed from 6.21 cm (3836) to -5.55 cm (73) took 11.5 sec 66mA (793mA peak) 23.7Vmin 327 AD/sec 460 ticks
1497878.068,HPITCH,N,TRACK: b: 94/0 a: 90/0 d: 2 o: -17
1497878.181,SUSR,N,Backing off pitch mass...
1497878.369,HPITCH,N,    Pitch control position = -5.56 cm, (AD#= 72 )
1497878.623,HPITCH,N,Pitch commanded from -5.56 cm (72) to -3.56 cm (711)...
1497879.979,HPITCH,N,-5.3 cm (ad: 166) .....MOTOR_DONE: ticks: 10 max 24v: 3.1mA avg 24v: 2.3mA minV 24v: 23.8V
1497883.122,SMOTOR,N,GC TICKS/TIME: 134/136259
1497883.217,HPITCH,N,done.
1497883.365,HPITCH,N,Pitch completed from -5.55 cm (73) to -3.59 cm (700) took 3.0 sec 235mA (292mA peak) 23.5Vmin 207 AD/sec 121 ticks
1497883.766,HPITCH,N,TRACK: b: 695/0 a: 698/0 d: 0 o: 2
```

**a.** At the prompt "Verify pitch SW limits and timeouts? [Y]," press ENTER to accept the default (Y). See Figure 7-19.

**b.** At the prompt "$PITCH_MIN: New value?[x]," verify that the value displayed matches the minimum pitch value on the Cal sheet (found in respective Seaglider notebook). See Figure 7-19.

**c.** Do one of the following:

- If the minimum pitch value is correct, press ENTER to accept the default answer.
- If the minimum pitch value at the prompt differs from the value on the trim sheet, make sure that the trim sheet is up to date. If the trim sheet is the latest version, enter the correct value, and then press ENTER.

**d.** At the prompt "$PITCH_MAX: New value?[x]," verify that the value on the screen matches the maximum pitch value on Seaglider's Cal sheet (found in respective Seaglider notebook). See Figure 7-19.

**e.** Do one of the following:

- If the maximum pitch value is correct, press ENTER to accept the default answer.
- If the maximum pitch value at the prompt differs from the value on the trim sheet, make sure that the trim sheet is up to date. If the trim sheet is the latest version, enter the correct value, and then press ENTER.

**f.** Do one of the following:

- If the default value for nominal pitch timeout appears (typical value should be 16 or 17), press ENTER.
- If the nominal value does not appear, enter the value **17** and press ENTER.

**42.** Check the roll motor and values.

**FIGURE 7-20. Checking Roll Motor**

```
1497883.880,SUSR,N,---- Checking roll motor ----
Verify roll SW limits and timeouts? [Y]
$ROLL_MIN
New value? [240.00]
$ROLL_MAX
New value? [3845.00]
1497894.742,SUSR,N,$ROLL_TIMEOUT: 15 OK
1497894.831,SUSR,N,Roll to port...
1497895.037,HROLL,N,Roll commanded from 0.14 deg (2600) to 35.34 deg (3845)...
1497896.403,HROLL,N,6.0 deg (ad: 2809) ...MOTOR_DONE: no ticks
1497898.345,SMOTOR,N,GC TICKS/TIME: 93/94947
1497898.438,HROLL,N,done.
1497898.590,HROLL,N,Roll completed from 0.14 deg (2600) to 35.22 deg (3841) took 2.2 sec 57mA (237mA peak) 23.6Vmin 551 AD/sec 90 ticks
1497898.994,HROLL,N,TRACK: b: 3817/0 a: 3828/0 d: -1 o: 13
1497899.110,SUSR,N,Roll to starboard...
1497899.329,HROLL,N,Roll commanded from 35.22 deg (3841) to -40.00 deg (1180)...
1497900.714,HROLL,N,29.2 deg (ad: 3628) ........MOTOR_DONE: ticks: 8 max 24v: 2.3mA avg 24v: 2.3mA minV 24v: 23.8V
1497905.405,SMOTOR,N,GC TICKS/TIME: 197/199250
1497905.500,HROLL,N,done.
1497905.660,HROLL,N,Roll completed from 35.22 deg (3841) to -39.86 deg (1185) took 4.7 sec 41mA (194mA peak) 23.7Vmin 571 AD/sec 186 ticks
1497906.061,HROLL,N,TRACK: b: 1216/0 a: 1212/0 d: 4 o: -27
1497906.180,SUSR,N,Roll to neutral...
1497906.364,HROLL,N,Roll commanded from -39.86 deg (1185) to 0.00 deg (2595)...
1497907.726,HROLL,N,-34.0 deg (ad: 1394) ....MOTOR_DONE: ticks: 7 max 24v: 3.1mA avg 24v: 2.3mA minV 24v: 23.8V
1497910.339,SMOTOR,N,GC TICKS/TIME: 113/115112
1497910.434,HROLL,N,done.
1497910.585,HROLL,N,Roll completed from -39.86 deg (1185) to 0.17 deg (2601) took 2.6 sec 63mA (253mA peak) 23.6Vmin 549 AD/sec 103 ticks
1497910.985,HROLL,N,TRACK: b: 2579/0 a: 2583/0 d: -6 o: 18
```

**a.** At the prompt, “Verify roll SW limits and timeouts? [Y],” press ENTER to accept the default (Yes). See Figure 7-20.

**b.** At the prompt “$ROLL_MIN: New value?[x],” verify that the value displayed matches the minimum roll value on the Cal sheet (found in respective Seaglider notebook). See Figure 7-20.

- If the minimum roll value is correct, press ENTER to accept the default.
- If the minimum roll value at the prompt differs from the value on the Cal sheet, make sure that the Cal sheet is up to date. If the Cal sheet is the latest version, enter the correct value, and then press ENTER.

**c.** At the prompt “$ROLL_MAX: New value?[x],” verify that the value displayed matches the maximum roll value on the Cal Sheet (found in respective Seaglider notebook). See Figure 7-20.

- If the maximum roll value is correct, press ENTER to accept the default answer.
- If the maximum roll value at the prompt differs from the value on the Cal sheet, make sure that the Cal sheet is up to date. If the Cal sheet is the latest version, enter the correct value, and then press ENTER.

**d.** Do one of the following:

- If the nominal value of 15 appears, then press ENTER.
- If the nominal value does not appear, enter the value 15 and press ENTER.

**43.** Check the VBD motor and values.

**FIGURE 7-21. Checking VBD Motor and Value**

```
1497911.104,SUSR,N,---- Checking VBD and valve ----
Verify VBD SW limits and timeouts? [Y]
$VBD_MIN
New value? [550.00]
$VBD_MAX
New value? [3959.00]
$VBD_TIMEOUT
New value? [720.00]
1497926.655,HVBD,N,   VBD lin pot AD counts: 632 544
1497926.809,HVBD,N,   VBD bladder position = 591.65 cc, (AD#= 588 )
1497927.077,HVBD,N,Bleed commanded from 591.65 cc (588) to 401.06 cc (1365)...
1497928.469,HVBD,N,589.4 cc (ad: 597 [647, 558]) ...................
1497939.244,HVBD,N,518.1 cc (ad: 888 [932, 852]) ...................
1497950.069,HVBD,N,449.6 cc (ad: 1167 [1208, 1135]) ..............MOTOR_DONE: ticks: 9 max 24v: 1076.4mA avg 24v: 123.2mA minV 24v: 23.7V
1497958.534,SMOTOR,N,GC TICKS/TIME: 1211/1213159
1497958.632,HVBD,N,done.
1497958.796,HVBD,N,Bleed completed from 591.65 cc (588) to 400.81 cc (1366 [1400, 1332]) took 30.0 sec 0mA (928mA peak) 23.8Vmin 25 AD/sec 1199 ticks
1497959.205,HVBD,N,TRACK: b: 1399/1332 a: 1399/1332 d: -34 o: -33
1497959.372,HVBD,N,   VBD lin pot AD counts: 1400 1332
1497959.528,HVBD,N,   VBD bladder position = 400.81 cc, (AD#= 1366 )
1497959.661,SUSR,N,Pumping 100cc on boost alone...
1497959.903,HVBD,N,Pump commanded from 400.81 cc (1366) to 500.89 cc (958)...
1497961.369,HVBD,N,401.3 cc (ad: 1364 [1399, 1327]) ...................
1497972.194,HVBD,N,424.4 cc (ad: 1270 [1315, 1223]) ...................
1497982.994,HVBD,N,448.4 cc (ad: 1172 [1216, 1126]) ...................
1497993.794,HVBD,N,472.9 cc (ad: 1072 [1116, 1027]) ...................
1498004.569,HVBD,N,497.2 cc (ad: 973 [1014, 929]) ...MOTOR_DONE: ticks: 9 max 24v: 3.8mA avg 24v: 3.1mA minV 24v: 23.8V
1498006.709,SMOTOR,N,GC TICKS/TIME: 1822/1824116
1498006.807,HVBD,N,done.
1498006.967,HVBD,N,Pump completed from 400.57 cc (1367) to 501.88 cc (954 [996, 913]) took 45.2 sec 111mA (1557mA peak) 23.6Vmin 9 AD/sec 1810 ticks
1498007.413,HVBD,N,TRACK: b: 1001/916 a: 1001/916 d: -43 o: -47
1498007.537,SUSR,N,Pumping up to maximum buoyancy...
1498007.781,HVBD,N,Pump commanded from 502.12 cc (953) to 600.98 cc (550)...
1498009.160,HVBD,N,502.9 cc (ad: 950 [990, 907]) ...................
1498019.960,HVBD,N,526.7 cc (ad: 853 [895, 810]) ...................
1498030.785,HVBD,N,550.2 cc (ad: 757 [798, 714]) ...................
1498041.535,HVBD,N,573.7 cc (ad: 661 [704, 617]) ...................
1498052.335,HVBD,N,596.6 cc (ad: 568 [609, 525]) ....MOTOR_DONE: ticks: 7 max 24v: 421.5mA avg 24v: 64.3mA minV 24v: 23.7V
1498054.996,SMOTOR,N,GC TICKS/TIME: 1841/1843146
1498055.095,HVBD,N,done.
1498055.256,HVBD,N,Pump completed from 502.12 cc (953) to 601.47 cc (548 [591, 505]) took 45.8 sec 545mA (2805mA peak) 23.3Vmin 8 AD/sec 1831 ticks
1498055.696,HVBD,N,TRACK: b: 594/507 a: 594/507 d: -44 o: -46
1498055.819,SUSR,N,Bleeding down 10 cc...
1498055.952,HVBD,N,   VBD lin pot AD counts: 589 502
1498056.105,HVBD,N,   VBD bladder position = 602.20 cc, (AD#= 545 )
1498056.373,HVBD,N,Bleed commanded from 602.20 cc (545) to 592.39 cc (585)...
1498057.741,HVBD,N,600.0 cc (ad: 554 [606, 513]) ..MOTOR_DONE: ticks: 8 max 24v: 1076.4mA avg 24v: 138.5mA minV 24v: 23.7V
1498059.376,SMOTOR,N,GC TICKS/TIME: 73/74931
1498059.468,HVBD,N,done.
1498059.631,HVBD,N,Bleed completed from 602.20 cc (545) to 592.14 cc (586 [631, 541]) took 1.5 sec 0mA (927mA peak) 23.8Vmin 26 AD/sec 62 ticks
1498060.025,HVBD,N,TRACK: b: 630/540 a: 631/540 d: -46 o: -45
0.260,SSURF,N,Initializing data file
0.542,SSURF,N,Initializing log file
```

**a.** At the prompt “Verify VBD SW limits and timeouts? [Y],” press ENTER to accept the default (Yes). See Figure 7-21.

**b.** At the prompt “$VBD_MIN: New value?[x],” verify that the value displayed matches the minimum VBD value on the Cal Sheet (found in respective Seaglider notebook). See Figure 7-21.

- If the minimum VBD value is correct, press ENTER to accept the default answer.
- If the minimum VBD value at the prompt differs from the value on the Cal sheet, make sure that the Cal sheet is up to date. If the Cal sheet is the latest version, enter the correct value, and then press ENTER.

**c.** At the prompt “$VBD_MAX: New value?[x],” verify that the value displayed matches the maximum VBD value on the Cal Sheet (found in respective Seaglider notebook). See Figure 7-21.

- If the maximum VBD value is correct, press ENTER to accept the default answer.

- If the maximum VBD value at the prompt differs from the value on the Cal sheet, make sure that the Cal sheet is up to date. If the Cal sheet is the latest version, enter the correct value, and then press ENTER.

**d.** Do one of the following:

- If the nominal value of 720 for VBD timeout appears, then press ENTER.
- If the nominal value does not appear, enter the value 720 and press ENTER.

**44.** Seaglider runs a self test on the sensors (both flight and science) and the pitch, roll and VBD systems and displays the data to the screen (Figure 7-22). With the exception of heading values fluctuating around zero when Seaglider is pointing north, the numbers should be non-zero. If the values look OK, press ENTER to continue.

**FIGURE 7-22. Checking sensors and data file creation**

```
2.182,SUSR,N,---- Checking sensors and data file creation ----
0.334,SSURF,N,Dive started Fri Apr 15 18:48:44 2011
 (1302893324)
5.483,SUSR,N,entering SD active
9.163,SSENSOR,N,A 5286ms 0.04m 19.2 #1
9.415,SUSR,N,entering SD passive
13.085,SSENSOR,N,P 9822ms -0.04m 19.0 #2
14.015,SUSR,N,entering SD active
17.690,SSENSOR,N,A 13819ms 0.02m 19.1 #3
17.958,SUSR,N,entering SD passive
21.634,SSENSOR,N,P 18366ms -0.03m 19.4 #4
22.558,SUSR,N,entering SD active
26.236,SSENSOR,N,A 22361ms 0.03m 18.3 #5
26.516,SUSR,N,entering SD passive
30.187,SSENSOR,N,P 26925ms -0.01m 18.7 #6
31.120,SUSR,N,entering SD active
34.800,SSENSOR,N,A 30924ms 0.11m 20.4 #7
35.092,SUSR,N,entering SD passive
38.765,SSENSOR,N,P 35499ms 0.06m 19.2 #8
39.695,SUSR,N,entering SD active
43.373,SSENSOR,N,A 39498ms 0.06m 18.3 #9
43.677,SUSR,N,entering SD passive
47.345,SSENSOR,N,P 44084ms 0.01m 19.0 #10
48.278,SUSR,N,entering SD active
51.957,SSENSOR,N,A 48081ms 0.13m 19.5 #11
52.273,SUSR,N,entering SD passive
55.944,SSENSOR,N,P 52681ms 0.07m 19.2 #12
56.874,SUSR,N,entering SD active
60.544,SSENSOR,N,A 56677ms 0.01m 18.6 #13
60.875,SUSR,N,entering SD passive
64.551,SSENSOR,N,P 61283ms 0.02m 19.6 #14
65.486,SUSR,N,entering SD active
69.164,SSENSOR,N,A 65289ms 0.07m 19.4 #15
version: 66.07
glider: 524
mission: 6
selftest: 170
start: 4 15 111 18 48 42
columns: rec,elaps_tms,depth,heading,pitch,roll,AD_pitch,AD_roll,AD_vbd,GC_phase,wlbbfl2vmt.600ref,wlbbfl2vmt.600sig,wlbbfl2vmt.Chlref,wlbbfl2v
data:
0 5286 4 192 -653 1326 701 2601 586 6 650 106 695 62 460 65 542
1 9822 -3 190 -655 1317 701 2601 586 6 650 106 695 62 460 65 542
2 13819 1 190 -652 1326 701 2601 586 6 650 93 695 62 460 65 541
3 18366 -2 194 -655 136 701 2601 586 6 650 100 695 62 460 65 541
4 22361 3 182 -653 1302 701 2601 586 6 650 108 695 62 460 65 541
5 26925 0 187 -653 1314 701 2601 586 6 650 99 695 62 460 65 541
6 30924 10 204 -653 1358 701 2601 586 6 650 98 695 62 460 65 541
7 35499 5 192 -6541323 701 2601 586 6 650 99 695 62 460 64 541
8 39498 6 183 -653 1302 701 2601 586 6 650 101 695 62 460 65 541
9 44084 0 189 -653 1311 701 2601 586 6 650 101 695 62 460 64 540
10 48081 12 194 -653 1331 701 2601 586 6 650 100 695 62 460 65 540
11 52681 7 19 -654 1323 701 2601 586 6 650 99 695 62 460 64 540
12 56677 0 186 -653 1311 701 2601 586 6 650 98 695 62 460 65 540
13 61283 2 196 -653 1334 701 2601 586 6 650 100 695 62 460 65 540
14 65289 7 193 -653 1323 701 2601 586 6 650 105 695 62 460 65 540
Hit CR to continue
```

**45.** The glider will now report its internal pressure and humidity.

Generally if the internal pressure is around 1 atmospheric pressure, 14.7 psi, a vacuum was not pulled. If a vacuum was pulled, the internal pressure should be approximately 5 psi below atmospheric pressure (or 10 psi). Humidity values would typically range between 30-40%.

If a vacuum was pulled, high internal pressure indicates a potentially dangerous problem. If the pressure inside the pressure hull rises 4psia above whatever the pressure is outside the pressure hull, the vent plug on the aft end cap opens to vent internal gasses.

If the prompt “Internal pressure is high. Continue self-test? [N]” appears, press ENTER to accept No (default). See Figure 7-23. Seaglider has failed the self test.

**Warning:** Before moving Seaglider, understand what is causing the high pressure and determine if it is safe to go near it.

**FIGURE 7-23. Internal Pressure and Humidity Sensors**

```
80.681,SUSR,N,---- Writing internal sensors to log ----
81.208,SUSR,N,internal humidity 1519 counts (34.008003RH)
81.416,SUSR,N,internal pressure 9.607714 psia
```

46. If the message “Sensor idx [x] not found in installed sensors!” appears, it indicates that there are sensors in slots [0] through [x-1]. Verify the number of installed sensors in the respective Seaglider notebook.

    For example, if the message Sensor idx 3 not found in installed sensors! appears, this indicates there are sensors in slot 0, slot 1, and slot 2, creating a total of 3 installed sensors (CT, DO, and Optics 1). If the number of sensors Seaglider says are installed differs from what is listed in the respective Seaglider notebook, the discrepancy needs to be understood and corrected. The self test result is a fail until the error is corrected.

47. The altimeter and transponder checks appear (Figure 7-24).If any error is present the self test fails. Determine the problem with the altimeter/transponder and correct before proceeding. Note that the altimeter/transponder does not always get a response in air.

**FIGURE 7-24. Checking Altimeter**

```
81.516,SUSR,N,---- Checking altimeter ----
81.971,HXPDR,N,ping response: R005.031ms

82.034,HXPDR,N,range was 3.773250 m
82.165,SUSR,N,---- Checking transponder ----
82.951,HXPDR,N,Tried 1 times to respond to command $T

83.054,SUSR,N,transponder reply ping count=-1
```

48. The glider will now report the list of targets from the currently loaded targets file and the settings from the currently loaded science file (Figure 7-25).

**FIGURE 7-25. Reporting Targets and Science Specifications**

```
83.159,SUSR,N,---- Reporting targets and science specifications ----
86.385,SNAV,N,      Target     Latitude  Longitude    Radius Depth Finish Timeout Next target Escp target
86.731,SNAV,N,      NW         4744.400  -12224.500    200.0m    0  -1  0.0 NE none
87.048,SNAV,N,  =>  NE         4744.400  -12223.700    200.0m    0  -1  0.0 SE none
87.365,SNAV,N,      SE         4743.000  -12223.700    200.0m    0  -1  0.0 SW none
87.682,SNAV,N,      SW         4743.000  -12224.500    200.0m    0  -1  0.0 NW none
87.998,SNAV,N,      T          4745.500  -12224.200    200.0m    0  -1  0.0 C1 none
88.315,SNAV,N,      C1         4744.200  -12223.600    200.0m    0  -1  0.0 B none
88.630,SNAV,N,      B          4743.100  -12223.600    200.0m    0  -1  0.0 C2 none
88.947,SNAV,N,      C2         4744.200  -12223.600    200.0m    0  -1  0.0 T none
89.822,SSENSOR,N,    Depth      Time       G&C Sensors
90.013,SSENSOR,N,    15.0m      5.0s     60.0s 1 (WL_BBFL2VMT: 1)
90.266,SSENSOR,N,    50.0m     10.0s    120.0s 1 (WL_BBFL2VMT: 1)
90.525,SSENSOR,N,   150.0m     15.0s    120.0s 1 (WL_BBFL2VMT: 1)
```

**49.** Next the glider will report the battery status/voltage as well as individual fuel gauges for all motors, sensors, and the main processor (Figure 7-26).

**FIGURE 7-26. Battery Status/Fuel Gauges**

```
85.615,SUSR,N,---- Reporting battery status ----

89.866,SPOWER,N,------ Fuel gauges (Cumulative Amp-secs) -------

                          cumulative      / since power up
90.151,SPOWER,N,             Pitch_motor:    17.29 amp-sec /    27.77 sec
90.351,SPOWER,N,              Roll_motor:     1.14 amp-sec /     6.85 sec
90.535,SPOWER,N,    VBD_pump_during_apogee:    45.86 amp-sec /     0.00 sec
90.761,SPOWER,N,    VBD_pump_during_surface:   565.23 amp-sec /   312.85 sec
90.917,SPOWER,N,                VBD_valve:     0.00 amp-sec /     0.00 sec
91.099,SPOWER,N,     Iridium_during_init:    34.50 amp-sec /     0.00 sec
91.286,SPOWER,N,    Iridium_during_connect:   122.36 amp-sec /     0.00 sec
91.904,SPOWER,N,     Iridium_during_xfer:   285.46 amp-sec /     0.00 sec
92.099,SPOWER,N,        Transponder_ping:     4.51 amp-sec /     1.00 sec
92.252,SPOWER,N,              GUMSTIX_24V:     0.00 amp-sec /     0.00 sec
92.428,SPOWER,N,      ------------- 24V total = 0.299 AmpHr -----------
92.631,SPOWER,N,                     GPS:     9.12 amp-sec /    65.16 sec
92.847,SPOWER,N,                     TT8:    40.84 amp-sec /    99.35 sec
93.050,SPOWER,N,                 LPSleep:     0.78 amp-sec /     5.90 sec
93.269,SPOWER,N,              TT8_Active:    29.28 amp-sec /   351.58 sec
93.492,SPOWER,N,            TT8_Sampling:   136.30 amp-sec /   179.32 sec
93.707,SPOWER,N,                TT8_CF8:    13.07 amp-sec /    32.80 sec
93.882,SPOWER,N,             TT8_Kalman:     1.73 amp-sec /     0.00 sec
94.102,SPOWER,N,          Analog_circuits:    28.83 amp-sec /   395.45 sec
94.255,SPOWER,N,             GPS_charging:     0.00 amp-sec /     0.00 sec
94.463,SPOWER,N,                 Compass:     7.84 amp-sec /    55.24 sec
94.615,SPOWER,N,                   RAFOS:     0.00 amp-sec /     0.00 sec
94.814,SPOWER,N,              Transponder:     0.68 amp-sec /     7.03 sec
94.967,SPOWER,N,                Compass2:     0.00 amp-sec /     0.00 sec
95.182,SPOWER,N,              WL_BBFL2VMT:    43.78 amp-sec /    49.64 sec
95.406,SPOWER,N,                   GPCTD:  3021.56 amp-sec /    65.44 sec
95.584,SPOWER,N,      ------------- 10V total = 0.926 AmpHr ------------
95.942,HBATT,N,24V batt pack voltage = 26.24V
96.077,HBATT,N,10V batt pack voltage = 10.85V
```

If running off external power, a message will appear that battery consumption is not being tracked.

**50.** Verify that the gauge is working properly:

The right-hand column is the cumulative number of the seconds each system has been on since the battery packs were replaced.

The gauge is working properly if the time (sec) in the right hand column (since power up) is reading mostly non-zero values while Seaglider is running on battery power.

If Seaglider is running on bench power, all values in the right hand column are near zero. If the right hand column reads all zeros while Seaglider is on battery power, the self test fails.

Determine the problem with the battery packs before proceeding.

**51.** The battery voltage display shows the amount of battery voltage left in the 10V and 24V battery packs.

If prompted "Battery voltage is low. Continue Self Test? [N]," do one of the following:

- If this self test is just before a planned launch, press ENTER to accept the default answer, No. The self test fails.
- Otherwise, type **Y** and press ENTER, if it is safe to do so.

**Note:** Often, if Seaglider is attached to the powered comms, the battery voltage appears out of range.

**52.** The capture file check settings are checked and the findings are displayed (Figure 7-27).

**FIGURE 7-27. Checking Capture Vector**

```
99.541,SUSR,N,---- Checking capture vector ----
99.648,SUSR,N,all capture settings at default values
```

**53.** Seaglider runs a self test on the persistor CF2 flashcard (Figure 7-28). If an error is detected, the self test fails. Determine the cause of the failure and correct before continuing.

**FIGURE 7-28. Checking Flash**

```
105.901,SUSR,N,---- Checking flash ----
No flash problems detected
```

**54.** When prompted to perform the communications check, press ENTER to accept the default answer Yes (Y).

Seaglider:

- Makes an Iridium call to the basestation.
- Downloads any files waiting for it on the basestation.
- Uploads the data files it has generated during the self test.

A message appears reporting the success or failure of the communication and transmission session (Figure 7-29). If the transmission fails, then the Iridium phone test fails and the self test fails. Reposition Seaglider to be sure its antenna has an unobstructed view of the sky and wait for it to make another phone call.

**FIGURE 7-29. Iridium Communications and Transmission Check**

```
371.501,SSURF,N,Trying call 1...
371.545,SSURF,N,Calling phone number: 6975087483006
373.654,HPHONE,N,initializing PSTN connection
396.647,HPHONE,N,Iridium signal strength: 5
396.812,HPHONE,N,Iridium geolocation: 41.729992 -71.595558 Wed Feb  9 16:08:54 2011
438.284,SSURF,N,Logged in...
439.544,SSURF,N,Sending cmd lsx -k -t150 cmdfile
443.668,SSURF,N,Receiving cmdfile...
444.934,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
448.779,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
450.307,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
451.837,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
453.368,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
454.898,SSURF,N,SOH/STX: 0x1 received (1), secSize= 128...
456.338,SSURF,N,EOT: 0x4 received (1), secSize= 128...

457.086,SSURF,N,Received cmdfile 709 bytes
457.517,SSURF,N,Transmission succeeded...
```

**55.** The pass/fail status of the self test is displayed. If the self test FAILED or ABORTED, the failures are listed. In the example in Figure 7-30, self test failed because:

- There were no bathymetric maps of the current location onboard Seaglider. Since Seaglider was on land, this error can be ignored.

Sometimes there will be an Iridium failure due to an inability to register at the beginning of the test because the time allotted for the phone registration is very short. If the Seaglider was able to successfully transfer data files at the end of the self test, this error can be ignored.

**FIGURE 7-30. Self Test Pass/Fail Status**

```
1015.236,SUSR,N,---- Self test FAILED or ABORTED! ----
1015.345,SUSR,N,1 failures noted
1015.425,SUSR,N,--> bathymetry maps failed
1015.521,SUSR,N,Restoring original settings...
```

**56.** Review the files (.log, .cap, .pvt, .dat) sent to the basestation via the Iridium call and the comm.log file for any warnings or errors.

At minimum, the pilot and preferably one other person should perform this review. The review is discussed under piloting tasks in Chapter 6, “Pre-Deployment Tasks.”

| IF... | THEN... |
|---|---|
| there are no warnings or errors seen in any of the files (expected outcome) | the test is complete. You can:<br>• Launch Seaglider (Chapter 6, “Pre-Deployment Tasks” on page 153).<br>• Perform a series of simulated dives (see “Simulated Dives” on page 219<br>• Wand off Seaglider and put it away. Go to the next step. |
| there is a problem during the testing, the final lines from the self test show that the self test failed or was aborted, or the data and log files were not transferred to the basestation | find the problem.<br>See Chapter 6, “Pre-Deployment Tasks” on page 153. |

**Note:** Whether the self test passes or fails, the Launch Menu appears.

(If your Seaglider is equipped with a GPCTD then continue with step 57, otherwise skip to step 58.)

**57.** IMPORTANT: You must turn the pump for the GPCTD back on, prior to deploying in water on a data-gathering mission.

**FIGURE 7-31. Loggers Menu**

```
├----- Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
Enter selection (1-5,CR): 2

------ Hardware Menu ------
*Motors and VBD
     1 [pitch   ]  Pitch control
     2 [roll    ]  Roll control
     3 [vbd     ]  VBD control
*Fixed Devices
     4 [super   ]  Supervisor
     5 [pressure]  Pressure sensor
     6 [compass ]  Compass (tcm2)
     7 [gps     ]  GPS
     8 [modem   ]  Modem (xmodem mode)
     9 [intpress]  Internal pressure
    10 [altim   ]  Altimeter
    11 [sensors ]  Sensors
    12 [loggers ]  Loggers
*Other
    13 [batt    ]  Batteries and fuel gauges
    14 [lowlevel]  Low-level hardware (IO,A-D,CF)
    15 [misc    ]  Miscellaneous (travel, timeouts, date/time)
    16 [develop ]  Developer tests
    CR)  Return to previous
Enter selection (1-16,CR): 12

------ Loggers menu ------
     1 [pc      ]  GPCTD
     2 [        ]  empty
    CR)  Return to previous
Enter selection (1-2,CR): 1
```

**a.** Press ENTER until you return to the main menu (Figure 7-31).

**b.** Press **2** and press ENTER to enter the Hardware Menu (Figure 7-31).

**c.** Press **12** and press ENTER to enter the Loggers Menu (Figure 7-31).

**d.** Press **1** and press ENTER to enter the GPCTD menu (Figure 7-31).

**FIGURE 7-32. Set MinCondFreq**

```
------ Logger device test menu ------
     1 [on      ]  Turn on
     2 [off     ]  Turn off
     3 [selftest]  Selftest
     4 [sample  ]  Report a sample
     5 [syncclk ]  Synchronize device clock to TT8
     6 [clock   ]  Read device clock
     7 [file    ]  Get file from device
     8 [action  ]  Execute logger action
     9 [config  ]  Show configuration
    10 [edit    ]  Edit configuration
    11 [direct  ]  Direct comms
    CR)  Return to previous
Enter selection (1-11,CR): 11
35.883,HGPCTD,N,RX = 10, TX = 11
36.579,SUSR,N,Starting breakable loop - Ctrl-Q to end

S>GetCD
<ConfigurationData DeviceType = 'SBE Glider Payload CTD' SerialNumber = '70100016'>
   <SampleDataFormat>HEX engineering</SampleDataFormat>
   <RS232ForceOn>yes</RS232ForceOn>
   <TxRealTime>no</TxRealTime>
   <SampleInterval>1</SampleInterval>
   <SampleMode>Continuous</SampleMode>
   <AutoRun>no</AutoRun>
   <SBE43>yes</SBE43>
   <MinCondFreq>10000</MinCondFreq>
   <CustomPumpMode>no</CustomPumpMode>
</ConfigurationData>
S>minCondfreq=3212.0
S>GetCD
<ConfigurationData DeviceType = 'SBE Glider Payload CTD' SerialNumber = '70100016'>
   <SampleDataFormat>HEX engineering</SampleDataFormat>
   <RS232ForceOn>yes</RS232ForceOn>
   <TxRealTime>no</TxRealTime>
   <SampleInterval>1</SampleInterval>
   <SampleMode>Continuous</SampleMode>
   <AutoRun>no</AutoRun>
   <SBE43>yes</SBE43>
   <MinCondFreq>3212.0</MinCondFreq>
   <CustomPumpMode>no</CustomPumpMode>
</ConfigurationData>
S>start
<!--start logging-->
S>    0.08, 15.4768, 0.00004
     0.09, 15.4771, 0.00004
     0.08, 15.4770, 0.00003
     0.07, 15.4778, 0.00003
     0.08, 15.4782, 0.00003
     0.08, 15.4788, 0.00003
     0.09, 15.4789, 0.00003
     0.09, 15.4798, 0.00003
     0.07, 15.4797, 0.00003
     0.08, 15.4799, 0.00003
     0.08, 15.4808, 0.00003
     0.08, 15.4811, 0.00003

<logging stopped/>

S>
------ Logger device test menu ------
     1 [on      ]  Turn on
     2 [off     ]  Turn off
     3 [selftest]  Selftest
     4 [sample  ]  Report a sample
     5 [syncclk ]  Synchronize device clock to TT8
     6 [clock   ]  Read device clock
     7 [file    ]  Get file from device
     8 [action  ]  Execute logger action
     9 [config  ]  Show configuration
    10 [edit    ]  Edit configuration
    11 [direct  ]  Direct comms
    CR)  Return to previous
Enter selection (1-11,CR): 2
```

**e.** Press **11** and press ENTER to go into direct serial communications with the GPCTD (Figure 7-32).

**f.** Press ENTER until a 'S>' prompt appears (Figure 7-32).

**g.** Type **MinCondFreq=X** where X is the number recorded when you followed the steps to disable the pump before and press ENTER. (Figure 7-32).

Below is some additional information on what the MinCondFreq should be set to.

**FIGURE 7-33. Example Excerpt from GPCTD Cal Sheet**

**SEA-BIRD ELECTRONICS, INC.**

13431 NE 20th Street, Bellevue, Washington, 98005-2010 USA

Phone: (425) 643 - 9866 Fax (425) 643 - 9954 Email: seabird@seabird.com

SENSOR SERIAL NUMBER: 0017
CALIBRATION DATE: 07-Dec-10

SBE GLIDER PAYLOAD CTD
CONDUCTIVITY CALIBRATION DATA
PSS 1978: C(35,15,0) = 4.2914 Siemens/meter

COEFFICIENTS:

g = -1.006405e+000
h = 1.720007e-001
i = -4.860079e-004
j = 6.696523e-005

CPcor = -9.5700e-008
CTcor = 3.2500e-006
WBOTC = 4.1244e-007

| BATH TEMP (ITS-90) | BATH SAL (PSU) | BATH COND (Siemens/m) | INST FREQ (Hz) | INST COND (Siemens/m) | RESIDUAL (Siemens/m) |
|---|---|---|---|---|---|
| 22.0000 | 0.0000 | 0.00000 | 2424.44 | 0.00000 | 0.00000 |
| 1.0000 | 34.5544 | 2.95570 | 4810.56 | 2.95570 | -0.00001 |
| 4.5000 | 34.5343 | 3.26073 | 4991.87 | 3.26073 | 0.00001 |
| 15.0000 | 34.4915 | 4.23596 | 5531.14 | 4.23597 | 0.00000 |
| 18.5000 | 34.4824 | 4.57883 | 5708.39 | 4.57882 | -0.00001 |
| 24.0000 | 34.4717 | 5.13300 | 5983.50 | 5.13300 | -0.00000 |
| 29.0000 | 34.4652 | 5.65125 | 6229.49 | 5.65125 | 0.00001 |
| 32.5000 | 34.4611 | 6.02101 | 6399.03 | 6.02101 | -0.00000 |

- Find the numbers circled above where Siemens/m is 0 indicating freshwater response
- The frequency above circled in red, 2424.44 Hz, is the number needed from the SBE cal sheet for this particular unit
- For saltwater and estuarine applications, SBE recommends the value circled above + 500 Hz. So, for the example above, the desired MinCondFreq would be 2924 Hz.
- For fresh/nearly fresh water application the typical recommended MinCondFreq is equal to the zero conductivity raw frequency + 5 Hz.

Please contact iRobot if there are any questions on what this value should be for your unit.

**h.** Type **GetCD** and press ENTER (Figure 7-32).
  **i.** Find the value labeled MinCondFreq and confirm it has been reset to the value it was before the first half of this process was run

**i.** Type **Start** and press ENTER (Figure 7-32).
  **i.** Confirm the pump is running if the value you entered was 0
  **ii.** The pump may not run if the value is non-zero

j. Type **Stop** and press ENTER to stop system and get prompt back. You will not see the word "Stop" displayed – but a message will appear that the "logging stopped" (Figure 7-32).

k. Press Ctrl-Q to exit direct communications with the GPCTD (Figure 7-32).

l. Once back at the GPCTD menu, type **2** and press ENTER to turn off the GPCTD

m. Press ENTER until you return to the main menu and wand off the Seaglider before continuing onto other tasks.

58. Turn off Seaglider:

a. Move the magnetic wand over the OFF symbol on the port side of Seaglider about 20" aft of the nose for at least 0.5 seconds. Make sure that the wand is touching the fairing.

b. While moving the wand over the OFF symbol press, or have someone else press, ENTER on the laptop keyboard.

c. Continue moving the wand around the OFF symbol and pressing ENTER until the laptop screen shows no response to pressing ENTER.

59. Disconnect the communication cable from Seaglider and the laptop and reinstall the dummy plug on Seaglider's communication connector.

60. If it is easier to store or move Seaglider in its cradle without the antenna mast attached, you can detach it (leave the cables attached to aft end cap). See the section "Detaching the Antenna Mast".

61. Stow Seaglider.

## *Simulated Dives*

Simulated dives are so named because they are done without Seaglider being in the water. They should be done outside so that the antenna mast has a clear view of the sky. Simulated pressure and pitch observations (**$SIM_W** and **$SIM_PITCH** respectively) are generated to allow test dives .

This is a valuable way to test the end-to-end data path, because the basestation is not only involved but has to deal with bi-directional file transmission multiple times. The operator can do as few, or as many, simulated dives as they would like. iRobot recommends that at least 5 are done.

To run simulated dives, the pilot needs to first make sure that files generated by Seaglider and transferred to the basestation at an earlier time (that is, during the recent self test) are preserved on the basestation, but not in the main directory.

Follow the steps below to check for files and, if any are present, move them to a self-describing directory.

1. On the basestation, enter cd **/home/sgNNN** (a.k.a.the “home directory”)

   where

   NNN = Seaglider’s serial number
2. If there are files present on Seaglider’s home directory from a past mission or test, enter the following:

   The longhand method to do this is to type:

   **/usr/local/basestation/movedata.sh -mission_dir <source_dir> -t target_dir>**

   to transfer the files.

   If you are in Seaglider’s home directory and want to move it to a subdirectory of that home directory, the shorthand method to move files is to type: **movedata.sh subdirectory_filename** and press ENTER.

   Some subdirectory naming convention examples are:

   - **Selftest_17May06** (after self test completed).
   - **Simulated dive_17May06** (after simulated dive completed).
   - **PortSusan_17May06** (after open water run in Port Susan).

   If the directory for that Seaglider does not exist on the basestation, see “Setting Up the Basestation Directories and Files” on page 35 for more information.
3. Edit the command file (.cmdfile) and possibly the targets file (.targets), the science file (.science) and the pagers file (.pagers) for the simulated dives.

These files are located on the basestation in the home directory of the Seaglider about to do the simulation.

To edit the files for the simulated dives:

**a.** Using any file editor, open the command file (cmdfile) and add or change the following parameters:

**$SIM_W,0.1**
**$SIM_PITCH,-20**
**$D_TGT,30**
**$T_DIVE,10**
**$T_MISSION,15**
**$KALMAN_USE,2**
**$SM_CC,150**
**$MAX_BUOY,5**
**$USE_BATHY,0**
**$T_RSLEEP,1**
**$C_VBD,*value provided in notebook or on CD***
**$C_PITCH, *value provided in notebook or on CD***
**$C_ROLL_DIVE, *value provided in notebook or on CD***
**$C_ROLL_CLIMB, *value provided in notebook or on CD***
**$GO**

**b.** Save the changes.

**c.** Using any file editor, open the targets file (targets) and put in one or two latitude/longitude values. See Appendix B, "Seaglider File Formats Manual" on page 275 for more detail on targets files.

The format of the file looks like the example below.

```
/ Simple targets
SEVEN   lat=4807.0 lon=-12223.0 radius=200 goto=SIX
SIX     lat=4806.0 lon=-12222.0 radius=200 goto=SEVEN
```

**d.** Save the changes.

**e.** Using any file editor open the science file (science) and make sure that the sensors installed on Seaglider are represented in the file. Also check the sampling rate. Since this test is usually done on battery power, the sampling rate is often reduced to cut down on battery usage.

The format of the file looks like the example below. See Appendix B, "Seaglider File Formats Manual" on page 275 for more detail on targets files.

**// Science for Port Susan**

/edit for Individual Seaglider w/: CT, SBE-43F oxy, WET Labs BB2FL-VMT

| /depth | time | sample | gcint |
|---|---|---|---|
| 10 | 4 | 122 | 60 |
| 40 | 4 | 144 | 120 |

**f.** Save the changes.

**g.** The simulated dives test is a good time to test the notification system provided by the .pagers file. The .pagers file controls the automatic email, pager and text message notification system. The file allows any of four types of messages (gps, alerts, recov, comp) to be sent to any valid email address or phone number. This service is run by the data conversion script, which is invoked by a Seaglider logout or disconnection.

Lines beginning with a # are comment lines and are ignored in processing. The gps message is sent after every connection and is the most recent GPS position. The alerts message is sent when the basestation has a problem converting a file or files. The file may be corrupt and have to be resent by Seaglider to the basestation using the pdoscmds.bat file (see Appendix C, "Extended PicoDOS® Reference Manual, v66.07" on page 303). The recov message is sent when Seaglider goes into recovery mode. This message includes the most recent GPS position and the recovery mode reason. The comp message tells the pilot what files completed processing on the basestation. An example .pagers file is below. An additional example is in Appendix B, "Seaglider File Formats Manual" on page 275.

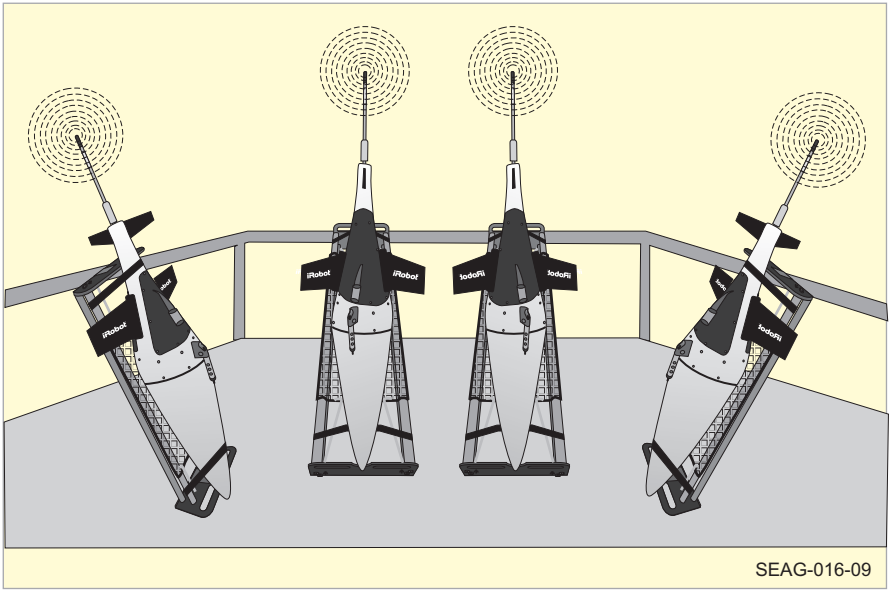
## Examples:
#
## Setting up an email account for gps, alerts, recov, comp
#someone@mydomain.edu,gps,alerts,recov,comp
#
## Setting up a phone for gps updates only
#6175551212@cingularME.com,gps
#
##Setting up an ATT phone for gps, alerts, recov, comp
#6175551212@cingularME.com,gps,alerts,recov

##Note the address for each phone service (that is, ATT, Verizon) differs

For a real contact address remove the #.

4. If Seaglider is not outside, assembled, communication cable attached and the antenna pointing up within 40° of vertical with an unobstructed view of the sky, do that now (Figure 7-34).

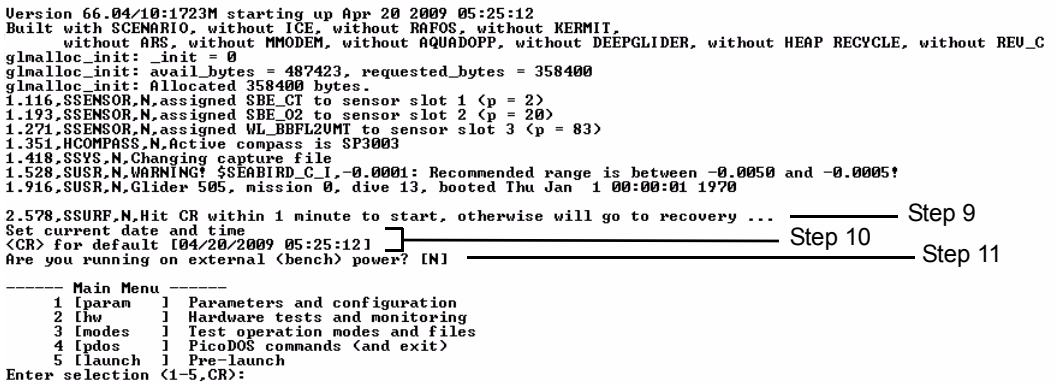
**FIGURE 7-34. Position of Seaglider During a Simulated Dive**




5. If the laptop is not already on and a terminal emulator running, do that now.
6. Open a screen capture or log file.
7. Connect the communication cable to the DB9 connector on the laptop (Figure 7-1).
8. Wand on Seaglider following the procedure on page 50.
9. When the output starts scrolling on the laptop, press ENTER within one minute (Figure 7-35).

   Follow the steps below, watch the process on the laptop and note if there are any errors or warnings during the test. The screen capture file can also be checked at the end of the test for warnings and errors.

**FIGURE 7-35. Seaglider Start-Up on the Laptop**


Version 66.04/10:1723M starting up Apr 20 2009 05:25:12
Built with SCENARIO, without ICE, without RAFOS, without KERMIT,
without ARS, without MMODEM, without AQUADOPP, without DEEPGLIDER, without HEAP RECYCLE, without REV_C
glmalloc_init: _init = 0
glmalloc_init: avail_bytes = 487423, requested_bytes = 358400
glmalloc_init: Allocated 358400 bytes.
1.116,SSENSOR,N,assigned SBE_CT to sensor slot 1 (p = 2)
1.193,SSENSOR,N,assigned SBE_O2 to sensor slot 2 (p = 20)
1.271,SSENSOR,N,assigned WL_BBFL2VMT to sensor slot 3 (p = 83)
1.351,HCOMPASS,N,Active compass is SP3003
1.418,SSYS,N,Changing capture file
1.528,SUSR,N,WARNING! $SEABIRD_C_I,-0.0001: Recommended range is between -0.0050 and -0.0005!
1.916,SUSR,N,Glider 505, mission 0, dive 13, booted Thu Jan 1 00:00:01 1970
2.578,SSURF,N,Hit CR within 1 minute to start, otherwise will go to recovery ...
Set current date and time
<CR> for default [04/20/2009 05:25:12]
Are you running on external (bench) power? [N]
Step 9
Step 10
Step 11
------ Main Menu ------
1 [param ] Parameters and configuration
2 [hw ] Hardware tests and monitoring
3 [modes ] Test operation modes and files
4 [pdos ] PicoDOS commands (and exit)
5 [launch ] Pre-launch
Enter selection (1-5,CR):

10. Set the current date and time (GMT).

    Any time within 12 hours is acceptable because Seaglider gets an accurate time from its first GPS fix and reset the internal clock (Figure 7-35).

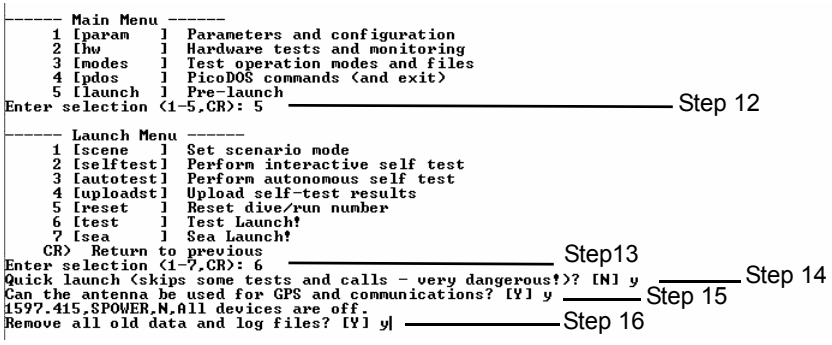
    **Note:** The format must be **mm/dd/yyyy hh:mm:ss**, with no missing values or extra spaces.

11. At the prompt "Are you running on external (bench) power?," type **Y** (for Yes) to indicate that you have an external power source (on powered comms cable); otherwise press ENTER to accept the default answer **N** (for No). (Figure 7-35)

    The Main Menu appears.

    Figure 7-36 shows the Main Menu for 1KA Seaglider testing.

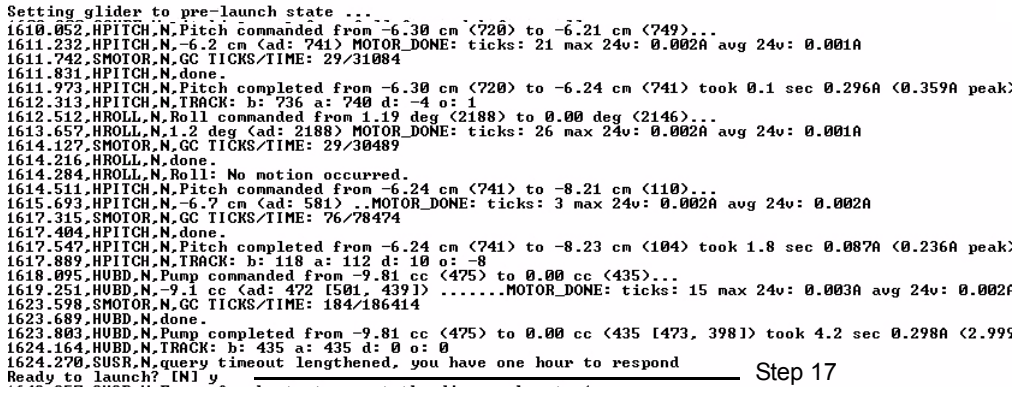
**FIGURE 7-36. Main Menu and Launch Menu**


------ Main Menu ------
1 [param ] Parameters and configuration
2 [hw ] Hardware tests and monitoring
3 [modes ] Test operation modes and files
4 [pdos ] PicoDOS commands (and exit)
5 [launch ] Pre-launch
Enter selection (1-5,CR): 5
Step 12
------ Launch Menu ------
1 [scene ] Set scenario mode
2 [selftest] Perform interactive self test
3 [autotest] Perform autonomous self test
4 [uploadst] Upload self-test results
5 [reset ] Reset dive/run number
6 [test ] Test Launch!
7 [sea ] Sea Launch!
CR) Return to previous
Enter selection (1-7,CR): 6
Step13
Quick launch (skips some tests and calls - very dangerous!)? [N] y
Step 14
Can the antenna be used for GPS and communications? [Y] y
Step 15
1597.415,SPOWER,N,All devices are off.
Remove all old data and log files? [Y] y
Step 16

12. Type **5**: Pre-Launch, and press ENTER. (Figure 7-36)
13. Type **6**: test Launch, and press ENTER. (Figure 7-36)
14. At the prompt “Quick Launch (skips some tests and calls - very dangerous if Seaglider is in the water!) [N],” type **Y** and then press ENTER**.** (Figure 7-36)
15. At the prompt “Can the antenna be used for GPS and Communications? [Y],” press ENTER to accept the default, Y (for Yes). (Figure 7-36)
16. At the prompt “Remove all old data and log files? [Y],” press ENTER to accept the default (Y for Yes). Beware, selecting ‘Yes’ removes all data files still on Seaglider from prior missions and tests. Be sure that those files are saved elsewhere BEFORE selecting ‘Yes’. If it is unknown whether the files have been saved, select N. (Figure 7-36)

    Seaglider performs pitch, roll, and pump (VBD maneuvers) and displays their movement on the laptop. (Figure 7-37)

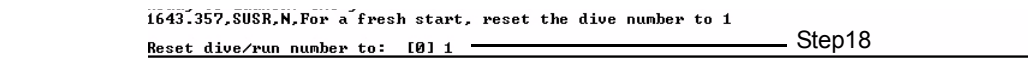
**FIGURE 7-37. Pitch, Roll, and Pump Maneuvers**


Setting glider to pre-launch state ...
1610.052,HPITCH,N,Pitch commanded from -6.30 cm (720) to -6.21 cm (749)...
1611.232,HPITCH,N,-6.2 cm (ad: 741) MOTOR_DONE: ticks: 21 max 24v: 0.002A avg 24v: 0.001A
1611.742,SMOTOR,N,GC TICKS/TIME: 29/31084
1611.831,HPITCH,N,done.
1611.973,HPITCH,N,Pitch completed from -6.30 cm (720) to -6.24 cm (741) took 0.1 sec 0.296A (0.359A peak)
1612.313,HPITCH,N,TRACK: b: 736 a: 740 d: -4 o: 1
1612.512,HROLL,N,Roll commanded from 1.19 deg (2188) to 0.00 deg (2146)...
1613.657,HROLL,N,1.2 deg (ad: 2188) MOTOR_DONE: ticks: 26 max 24v: 0.002A avg 24v: 0.001A
1614.127,SMOTOR,N,GC TICKS/TIME: 29/30489
1614.216,HROLL,N,done.
1614.284,HROLL,N,Roll: No motion occurred.
1614.511,HPITCH,N,Pitch commanded from -6.24 cm (741) to -8.21 cm (110)...
1615.693,HPITCH,N,-6.7 cm (ad: 581) ..MOTOR_DONE: ticks: 3 max 24v: 0.002A avg 24v: 0.002A
1617.315,SMOTOR,N,GC TICKS/TIME: 76/78474
1617.404,HPITCH,N,done.
1617.547,HPITCH,N,Pitch completed from -6.24 cm (741) to -8.23 cm (104) took 1.8 sec 0.087A (0.236A peak)
1617.889,HPITCH,N,TRACK: b: 118 a: 112 d: 10 o: -8
1618.095,HVBD,N,Pump commanded from -9.81 cc (475) to 0.00 cc (435)...
1619.251,HVBD,N,-9.1 cc (ad: 472 [501, 439]) .......MOTOR_DONE: ticks: 15 max 24v: 0.003A avg 24v: 0.002A
1623.598,SMOTOR,N,GC TICKS/TIME: 184/186414
1623.689,HVBD,N,done.
1623.803,HVBD,N,Pump completed from -9.81 cc (475) to 0.00 cc (435 [473, 398]) took 4.2 sec 0.298A (2.995
1624.164,HVBD,N,TRACK: b: 435 a: 435 d: 0 o: 0
1624.270,SUSR,N,query timeout lengthened, you have one hour to respond
Ready to launch? [N] y
Step 17

17. At the prompt "Ready to launch? [N]," type **Y** (for Yes), and then press ENTER. (Figure 7-37)
18. At the prompt "Reset dive/run number to:[0]," type **1** or a number of your choosing and then press ENTER. (Figure 7-38)

    **Note:** If you chose not to remove old data and log files, select a number that does not overlap those already present.

**FIGURE 7-38. Reset Dive/Run Number**




19. Verify that Seaglider logs in. Then watch as the parameters (cmdfile), science and targets files are uploaded.

    While the files are uploading, the laptop display looks like Figure 7-39.

**FIGURE 7-39. File Upload to Seaglider**

Step 19

```
132.383,SSURF,N,Logged in...
134.022,SSURF,N,Sending cmd lsx -k -t150 cmdfile
137.437,SSURF,N,Receiving cmdfile...
138.592,SSURF,N,SOH/STX: 0x1 received, secSize= 128...
140.213,SSURF,N,SOH/STX: 0x1 received, secSize= 128...
141.653,SSURF,N,SOH/STX: 0x1 received, secSize= 128...
143.182,SSURF,N,SOH/STX: 0x1 received, secSize= 128...
144.712,SSURF,N,SOH/STX: 0x1 received, secSize= 128...
146.245,SSURF,N,EOT: 0x4 received, secSize= 128...

147.578,SSURF,N,Received cmdfile 573 bytes
149.040,SSURF,N,Transmission succeeded...
149.128,HTI8,N,Updating parameter $SEABIRD_T_G to 0.0042812955
149.220,HTI8,N,Updating parameter $SEABIRD_T_H to 0.00061992655
149.313,HTI8,N,Updating parameter $SEABIRD_T_I to 2.2443779e-05
149.406,HTI8,N,Updating parameter $SEABIRD_T_J to 2.3337107e-06
149.499,HTI8,N,Updating parameter $SEABIRD_C_G to -9.9470167
149.588,HTI8,N,Updating parameter $SEABIRD_C_H to 1.1498449
149.681,SUSR,N,WARNING! $SEABIRD_C_I,-0.0001: Recommended range is between -0.0050 and -0.000
149.805,HTI8,N,Updating parameter $SEABIRD_C_I to -9.315474e-05
149.898,HTI8,N,Updating parameter $SEABIRD_C_J to 0.00010132478
149.990,HTI8,N,Updating parameter $PRESSURE_SLOPE to 0.00011485742
150.084,HTI8,N,Updating parameter $VBD_MIN to 435
150.158,HTI8,N,Updating parameter $VBD_MAX to 3963
```

20. During the test dives:
    a. Monitor the activity.

       Files should be successfully transmitted bi-directionally between Seaglider and the basestation. If a transfer fails during the test, let the test continue through the five dives. Perhaps the Iridium satellite was shaded at the time. However, after the dives are complete, investigate the failures and make sure that there is a full understanding of the mechanism. Repeat the simulated dives, if necessary.

    b. When Seaglider is starting its 5th dive, change the directive in the cmdfile on the basestation from **$GO** to **$QUIT**. Seaglider uploads the file at the next 'surfacing' and remains on the surface.

    c. Wand off Seaglider after it is parked on the 'surface' following the fifth dive and all files have been transferred to the basestation. Press ENTER on the laptop several times. If information displays, Seaglider is not off. Wand off again. If nothing displays after you repeatedly press ENTER, Seaglider is off. As a double check, leave Seaglider outside for 5-10 minutes and monitor the basestation or laptop for any Seaglider activity. If Seaglider calls in, it is not off and burning battery power! Wand it off again and monitor to make sure it is off.

21. Disconnect the communication cable from Seaglider and the laptop and reinstall the dummy plug on Seaglider's communication connector.

22. If it is easier to store or move Seaglider in its cradle without the antenna mast attached, detach it (leave the cables attached to aft end cap). See ".pagers file" on page 172.

23. After the test dives:

a. Review all of the files generated by Seaglider and the basestation during the simulated dives. See Chapter 9, “Files for Operations” on page 237 for an explanation of each file. Check the .log and .cap files for errors and warnings. Check the science sensor data in the .dat, .asc, and .eng file. Check the comm.log files for communication exchanges between Seaglider and the basestation. If anything looks suspect, investigate and understand the cause and correct before continuing.

b. Copy these files into a date-encoded sub-directory of Seaglider’s home directory (/home/sgXXX/SimulatedDive_DDmonthYY).

The longhand method to do this is to type:

**/usr/local/basestation/movedata.sh -mission_dir <source_dir> -t target_dir>**

to transfer the files.

If you are in Seaglider’s home directory and want to move it to a subdirectory of that home directory, the shorthand method to move files is to type:

**movedata.sh subdirectory_filename**

and press ENTER.

The UNIX ‘mv’ command can also be used to move data from the home directory.

## ARGOS tag

Prior to deployment user parameters must be programmed into the SPOT5 Argos tag to properly fit the application needs for a Seaglider mission. User parameters are programmed into the SPOT5 via a Windows-based program provided by Wildlife Computers called SPOT5Host. The SPOT5 is connected to a PC via the USB-Blue, available from Wildlife Computers only. SPOT5Host synchronizes communications with the POST5’s on-board software. It sends and receives commands to the SPOT5. Parameters are downloaded from SPOT5 to the PC. SPOT5Host allows modification of those parameters. Once the parameters are set, they can be uploaded to the SPOT5.

For directions on programming SPOT5 user parameters, see the iRobot, Argos SPOT5 Tag Setup and Installation “document# 4335290”

CHAPTER 8

# *Tuning Your Seaglider*

This chapter describes tuning your Seaglider.

## *Tuning Your Seaglider*

To tune your Seaglider:

**Pilot:** After Seaglider has surfaced from its first dive and transmitted the data files to the basestation, review the .log file and the capture file, if it is sent (recommended at the beginning of a mission) and analyze the data. When reviewing the .log file, look at the time in the GPS line. Is it current? Is the GPS fix believable? Are the internal pressure and humidity values within range? Are there any retries or errors? Plot the data from the .eng and .log files using the MATLAB scripts provided on the CD shipped with Seaglider.

Look at the main plot (Figure 8-1 on page 231). Interpretation of the plot is a major subject in training class.

- Does the dive velocity match the desired velocity?
- Is the buoyancy (VBD) set properly for water density at the prescribed depth?

- Is Seaglider pitched properly? If the pitch setting is incorrect, it affects the vertical velocity. If the VBD and the pitch are both set correctly, Seaglider has a symmetrical dive pattern.
- Is Seaglider flying straight? This is dependent upon the roll centers being properly set. Normally, adjustments to Seaglider's flight path are made in the order VBD and Pitch during the early, shallow dives, then, the roll is adjusted during the deeper dives when Seaglider has time to glide between motor movements. If any changes to the VBD, Pitch and roll centers are needed, the rule of thumb is to only adjust half the amount suggested by the plot regression for 'Implied.'

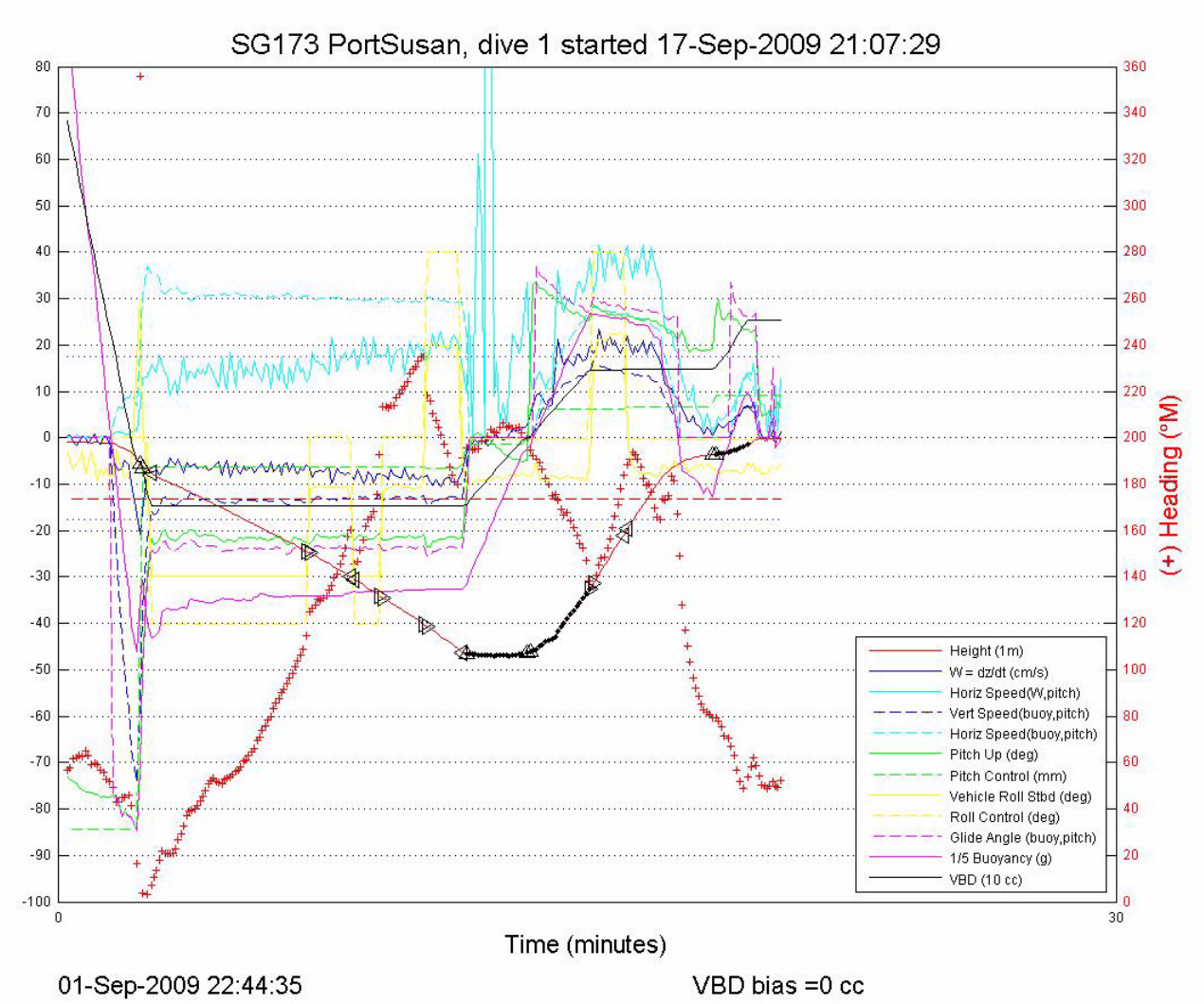
  Using this approach look at the first plot (an example is below) and compare when the VBD (thin black line) and the vertical speed (W) (royal blue line) cross zero.
- If they cross zero at the same time the VBD setting is good and should be left alone.
- If the VBD line crosses zero before the W line does, Seaglider is too heavy and the C_VBD A/D counts should be reduced.
- If W crosses zero before the VBD line crosses zero, Seaglider is too buoyant and the C_VBD A/D counts should be increased.

In the example below the W line crosses zero before the VBD line so Seaglider is too buoyant.

To figure out how much to increase the C_VBD by in the example below, determine the number of units that separates the W line from the VBD line where the W crosses zero. In the example it is approximately 4 units. Multiply that value by the number of cc's in parentheses next to VBD in the legend. In the example it is 10cc. So 4* 10cc = 40 cc. Now convert the 40cc to A/D counts using the conversion factor 4.0767 A/D counts/cc to get an A/D counts change of approximately 160.

Following the rule of thumb to only make half of the adjustment at a time, the C_VBD would be changed from the original A/D counts of 3276 + 80 = 3356 and the parameter value would be changed in the cmdfile to read **$C_VBD,3356**.

**FIGURE 8-1. Main Plot**

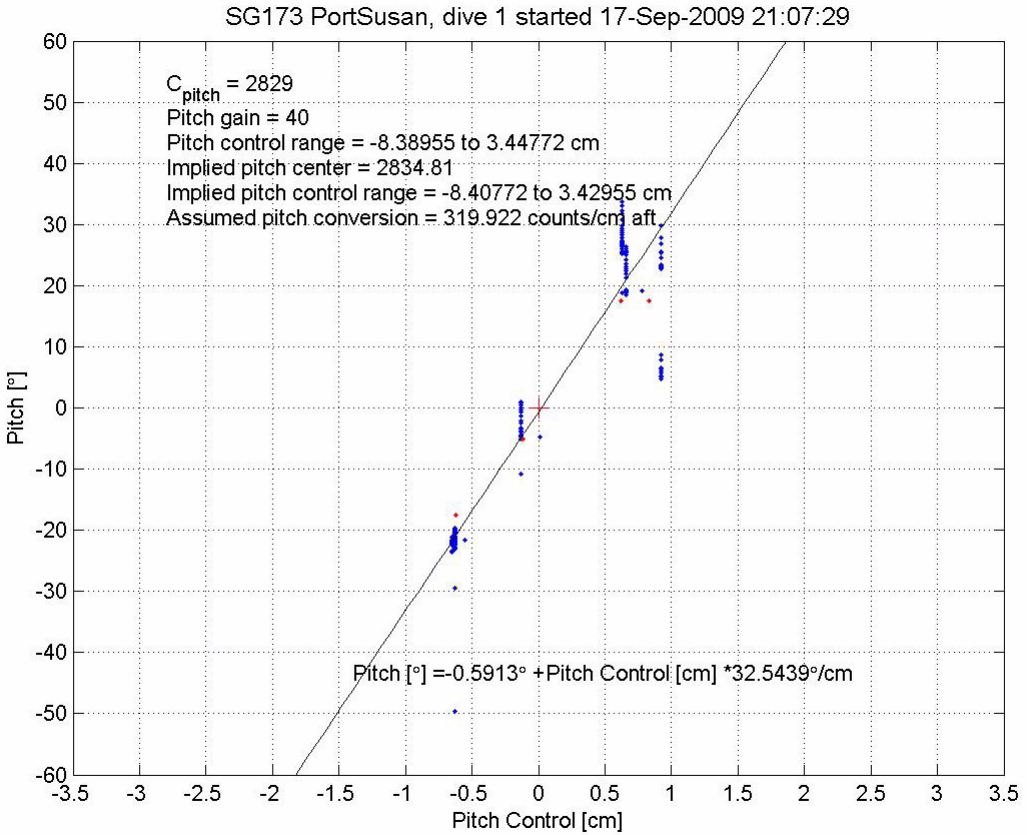



- Look below at the pitch plot (Pitch Control versus Pitch (deg)) and the regression analysis located on the plot, showing what the present pitch center **$C_PITCH** is and what the recommended pitch center is (Figure 8-2).

  Following the rule of thumb for making changes to the parameter value: the present **$C_PITCH** value is 2829 and the Implied (desired) **$C_PITCH** is

2835 so the correction put into the cmdfile would be **$C_PITCH,2832**. This change in the **$C_PITCH** parameter value is very small and is actually in the 'noise.' Generally, if the difference in A/D counts between the actual and implied **$C_PITCH** values is less than 15-20 counts no adjustment is made. Additionally, at the bottom of the plot is the calculated pitch gain. The initial **$PITCH_GAIN** value should be adjusted by half of the difference between it and the calculated pitch gain (rounded to the nearest whole number) and loaded into Seaglider via the cmdfile transfer.

**FIGURE 8-2. Dive 1 Pitch Plot**




- Inspect the science data plots and data files. Is the data believable?

If there is a problem with the science data Seaglider may have to be recovered.

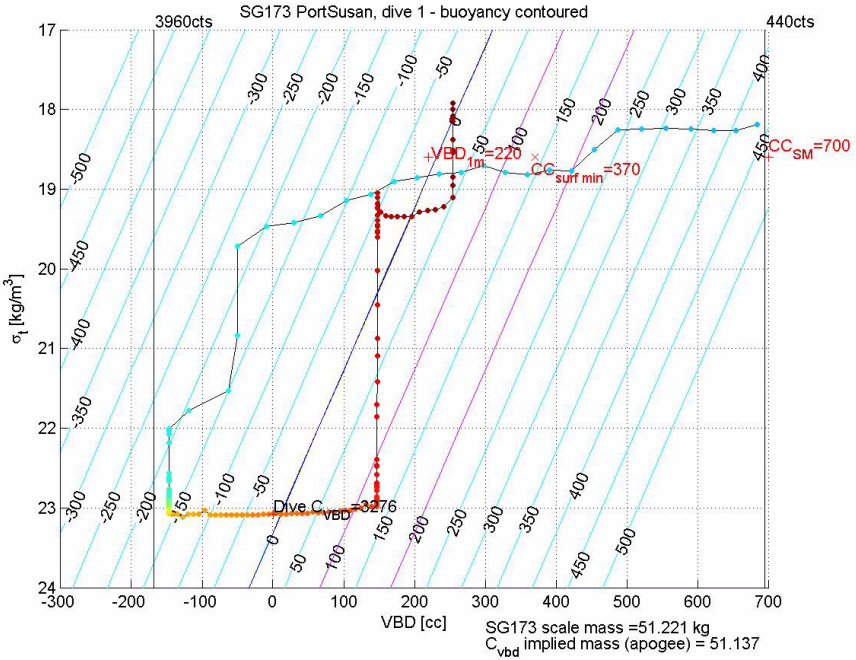
If the science data looks good and the cmdfile has been updated with the latest **$C_VBD, $C_PITCH and $C_PITCH_GAIN,** it is time to start the second dive.

- Check that the new **$C_VBD, $C_PITCH and $C_PITCH** parameters have been transferred to Seaglider and are correct. If they are, change the directive in the cmdfile from **$QUIT** to **$RESUME.** Leave the dive parameters the same as for the first dive. When Seaglider calls in again after **$T_RSLEEP** it picks up the new directive and begin the second dive.
- The pilot should communicate with the field team what is being done and the field team should let the pilot know when Seaglider leaves the surface on the second dive. The pilot can then change the directive in the cmdfile from **$RESUME** to **$QUIT** so that Seaglider remains on the surface after the second dive.
- When Seaglider surfaces after the second dive, the pilot should check the log file, plot the data, review the plots, especially the VBD and pitch plots, and make any needed adjustments to the **$C_VBD** and **$C_PITCH** A/D counts and **$PITCH_GAIN**. If the pilot feels comfortable with Seaglider, the field team can be dismissed. It is also the pilot's decision whether to have Seaglider do another shallow dive or whether the next dive should be a 100-200m dive. If the depth is increased remember to edit **$D_TGT, $T_DIVE and $T_MISSION**. Remember to check what directive is in the cmdfile and change it, if necessary.

  By the third or fourth dive, the value for **$SM_CC** set at the beginning should be revisited. Initially **$SM_CC** is set conservatively to make sure Seaglider gets back to the surface. However, the $\sigma_T$ versus VBD (cc) plot shows what the **$SM_CC** setting currently is and what the setting could be changed to.

  An example plot is below. The current setting is located in the upper left corner of the plot ($CC_{SM}$=700) and the statement $CC_{surf\ min}$ = 370 shows what the **$SM_CC** value could be changed to, to get the full antenna mast out of the water.

**FIGURE 8-3. $SM_CC Values**




After several 100m or greater depth dives look at the roll plot (Roll Control (deg) versus Roll (deg)) and the roll regressions on the plot and make any necessary corrections. Remember the rule of thumb for making changes and only make half of the correction desired at a time.

An example of a roll plot is shown in Figure 8-4 on page 235. In the example, **$C_ROLL_CLIMB** is currently set at 2000 A/D counts. The implied (desired) **$C_ROLL_CLIMB** A/D counts is approximately 2350. Taking the difference between the two values and dividing by two the **$C_ROLL_CLIMB** value should be changed to 2175.

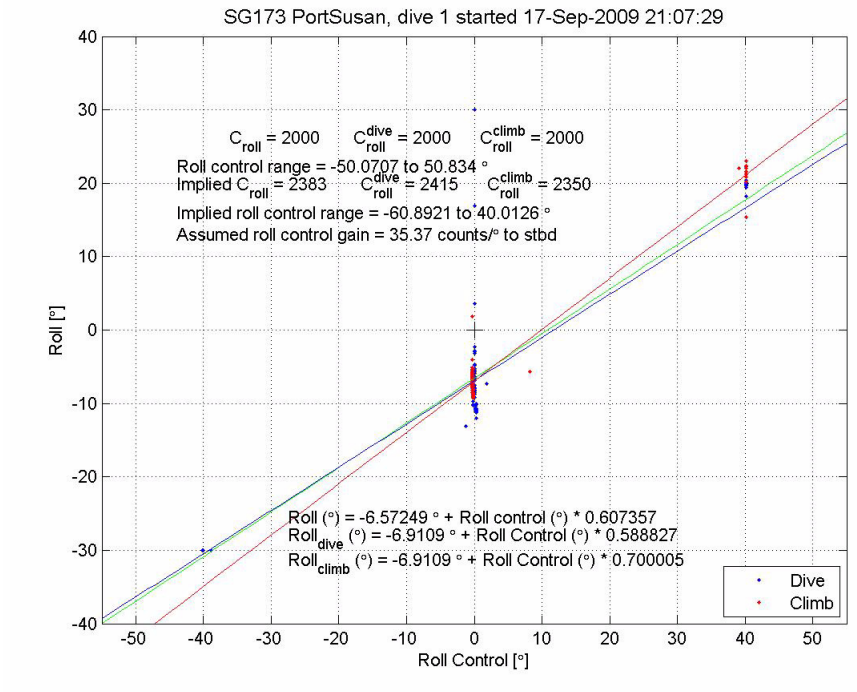
Follow the same procedure for determining the correction to the **$C_ROLL_DIVE** value.

As the trimming becomes less drastic, the need to park Seaglider on the surface after a dive with the **$QUIT** command, while changes to the parameter values are made and then to issue the **$RESUME** directive to continue diving, becomes less critical.

At this point, if Seaglider is diving, the directive should be changed from **$RESUME** to **$GO** and any changes made to a parameter are picked up at the next surfacing.

Remember that with the **$GO** directive, if Seaglider encounters a problem and has to surface unexpectedly, it remains at the surface until the pilot replaces **$GO** with **$RESUME**. DO NOT leave **$RESUME** in your cmdfile longer than necessary. It may lead to the loss of your Seaglider.

**FIGURE 8-4. Roll Center Values**




Continue to monitor the plots and the log files each time Seaglider surfaces until the trimming looks good and Seaglider is diving to the desired mission depth. As the pilot becomes comfortable with how Seaglider is flying, plot and file checks can be reduced, if desired, to several times a day.

CHAPTER 9

# *Files for Operations*

To be a successful pilot, gathering high quality data during the mission and bringing Seaglider home safely, one needs to understand the intricacies of the files that must be generated to fly Seaglider as well the contents of the files that are generated by Seaglider during the mission.

The following topics are covered:


- "Files Placed on the Basestation by the Pilot" on page 237
- "Factory-Generated File" on page 246
- "Seaglider-Generated Files" on page 246
- "Basestation-Generated Files" on page 248
- "Files Stored on Seaglider" on page 253


## *Files Placed on the Basestation by the Pilot*

To command and control Seaglider, the pilot interacts with four files on the basestation:

- Command file (cmdfile)
- Targets file (targets)

- Science file (science)
- PicoDOS Commands Batch File (pdoscmds.bat)

These files are sent to Seaglider during a surfacing. However, unless there are changes to the target, science or pdoscmds.bat files, they do not have to be sent to Seaglider at each surfacing.

Only the cmdfile must be sent to Seaglider at each surfacing. After one of these files has been uploaded to Seaglider, the number of the dive is appended to the file name on the basestation. If Seaglider calls in to the basestation more than once while at the surface, the files are uploaded each time if they are present on the basestation and after each upload not only is the dive number appended as a .number, but the upload number is appended as a .number to the dive number.

The following example shows when the file is uploaded and how it is named:

- A cmdfile is uploaded to Seaglider.
- The basestation renames the file cmdfile.1.
- Seaglider is still at the surface, calls in again.
- The cmdfile is uploaded to Seaglider.
- The basestation renames the file cmdfile1.1.
- Seaglider is still at the surface, calls in again and the cmdfile is uploaded to Seaglider.
- The basestation renames the file cmdfile1.2.
- Seaglider then dives.
- At the next surfacing a cmdfile is uploaded to Seaglider.
- After the upload, the basestation renames the file as cmdfile.2.

The same process is used for the targets, science and pdoscmds.bat files. More details on these files is below.

A fifth file, the sg_calib_constants.m file is generated at the factory. It is not sent to Seaglider. Instead, it is used shore-side by MATLAB to process the .eng files and .log files and produce Seaglider data plots. See “Calibration File” on page 246.

### Command File

The cmdfile is sent to Seaglider at each surfacing. The cmdfile configuration is:

- The listing of the parameters (one per line).
- A comma.
- The new value for each, with no space between the comma and the value.
- The last line of the file is a directive (required).

The command file may contain only the directive. So, if no changes are needed in the cmdfile and Seaglider is to continue diving after the next surfacing, it can contain the only the directive **$GO**.

If no changes are needed in the cmdfile and Seaglider is to remain at the surface after the next surfacing, the command file can only contain the directive **$QUIT**.

If parameters are left in the cmdfile from dive to dive, they are sent each time Seaglider surfaces even though the value of the parameter has not changed between dives.

An example of a cmdfile with parameters and a directive is below.

```
$D_TGT,150
$T_DIVE,50
$T_MISSION,60
$SM_CC,680
$C_ROLL_DIVE,2388
$C_ROLL_CLIMB,2321
$USE_BATHY,-4
$ALTIM_SENSITIVITY,4
$QUIT
```

### Targets File

Seaglider navigates using either heading or targets (waypoints). If heading is used, it is turned on in the cmdfile **($HEADING**,value in degrees). See “$HEADING” on page 117.

If Seaglider is moving from location to location using targets (latitude and longitude) also known as waypoints, it is getting this information from the targets file.

An example of a targets file, and a description of each column, is below. Another example can be found in Appendix B, “Seaglider File Formats Manual” on page 275.

Example targets file

```
/ Sample North Carolina Coast targets
NCSPONE   lat=3357.4   lon=-7623.5   radius=200   goto=WPONE     timeout=1
WPONE     lat=3356.0   lon=-7625.2   radius=200   goto=WPTWO     timeout=1
WPTWO     lat=3353.9   lon=-7625.2   radius=200   goto=WPTHREE   timeout=1
WPTHREE   lat=3352.2   lon=-7621.4   radius=200   goto=WPFOUR    timeout=1
WPFOUR    lat=3356.6   lon=-7623.8   radius=200   goto=NCSPONE   timeout=1
```

| Column | Description |
|---|---|
| 1 | Name of the target Seaglider is heading to |
| 2 | Latitude of the target Seaglider is heading to in degrees, minutes, decimal minutes |
| 3 | Longitude of the target Seaglider is heading to in degrees, minutes, decimal minutes |
| 4 | Distance in meters Seaglider can be from the target and still be considered having reached the target |
| 5 | Name of the next target to go to after it has reached the present target |
| 6 | After the timeout is exceeded the glider will proceed to the next waypoint as though it had just reached the current waypoint. The timeout if a floating point value and is specified in days. |

The file above contains all of the fields necessary to direct Seaglider to targets. There are also three optional fields, which can be added as columns in the targets file. They are:

```
escape=WPONE      depth=100      finish=90
```

where:

- The escape target specifies what target to move to if Seaglider has been unable to navigate for a specified length of time (stuck under an object). The escape target must be a valid named target.

- The depth allows the pilot to define the target as a depth.
- The finish specifies a direction (degrees), and establishes a finish line through the target, perpendicular to the direction specified. The target is considered achieved when the difference between the bearing to the target and the finish direction is greater than 90 (or less than -90) degrees.

**Example 1**

Finish direction of 90 specifies a north-south finish line drawn through the target; the target is achieved when Seaglider is east of the line.

**Example 2**

Finish direction of 180 specifies an east-west finish line; target is achieved when Seaglider is south of the line. A value of -1 or no specification of finish means that no finish line is tested.

For new missions, Seaglider heads to the first waypoint listed, in this case the one called NCSPONE. To change the waypoints file during a mission the pilot can upload a new targets file when Seaglider surfaces. However, use caution when doing this.

- If an active target name from the old targets file is in the new targets file, Seaglider retains that active target.
- If an active target is not in the new targets file, then Seaglider goes to the first target in the list.
- If there is no valid targets file on Seaglider, it uses the default target (**$TGT_AUTO_DEFAULT, $TGT_DEFAULT_LAT, $TGT_DEFAULT_LON**) specified in the cmdfile.

  Though specification of a default target is optional, it is highly recommended that you enter the latitude and longitude coordinates of a location where you can recover Seaglider in the event that it loses its targets file.

### ALI Sensors

ALI sensors are controlled via 5 different parameters as listed below. The $xx is the prefix for a given sensor which is defined by the command prefix in the .cnf file. For example, the prefix for the GPCTD is PC.

| | |
|---|---|
| **$xx_RECORDABOVE** | depth (in meters) above which the logger is on. A value of 0 turns the sensor off.<br>Minimum Value: 0<br>Maximum Value: 1000 |
| **$xx_PROFILE** | which half of the profile to run the logger 0=none, 1=dive, 2=climb, 3=both |
| **$xx_XMITPROFILE** | which half of the profile to transmit data back via Iridium 0=none, 1=dive, 2=climb, 3=both |
| **$xx_UPLOADMAX** | the maximum file size (in bytes) limit on uploaded files.<br>Minimum Value: 0, no data uploaded. |
| **$xx_STARTS** | running count of the number of sensor restarts that occur during a mission |

Additional parameters can be defined using param-x=, param-y=and param-z=. For example, specifying param-x=RATE and specifying $xx_RATE,1.2 in the cmdfile will cause the interpreter to expand %x in a command string to the value 1.2.

Lines in script files (x, y, and z) are fully interpreted. The script mechanism is intended to allow complex, multi-command interactions or to provide a pilot control of a device during a mission. For example, a start command might become:

start=%X
script-x=device.bat

where the contents of device.bat are:

%r%pvalue1=17.0%r
%pvalue2=0.05%r
%pstart%r

The pilot could change the values of the parameters value1 and value2 by creating a new file device.bat on the basestation. That file will be transferred to the glider CF card during a call and the logdev driver will send the commands changing those values the next time it issues a start command.

## Science File

The science file controls how often any OSI science sensors and the G&C sensors are sampled. The sample rate for each science sensor is independent of the other science sensors. The sampling protocol is specified by depth bins and each bin can have a different sampling protocol. The practical lower limit on sampling is 4 seconds.

If only the conductivity and temperature sensors are sampled, it can be possible to sample every 4 seconds, but with the oxygen and BBFL2 or BB2FL optical sensors also being sampled, 5 seconds is the lower limit. The science file also provides the ability to turn off sensors, or only energize them every $n^{th}$ sample, in a given depth range (or ranges).

An example science file is below. Another example can be found in Appendix B, “Seaglider File Formats Manual” on page 275.

```
// Science for North Carolina Coast
/edit for Individual Seaglider w/: CT, SBE-43F oxy, WET Labs BB2FL-
VMT
/depth          time            sample          gcint
50              5               111             30
200             5               121             60
300             5               103             120
```

| Column | Description |
|---|---|
| 1 | The depth bin in meters for that sampling protocol |
| 2 | Base sampling interval in seconds |
| 3 | Multipliers for each of the three sensors listed on line 2 of the file, in the order listed, applied to the base sampling interval |
| 4 | Guidance and Control (G&C) sampling interval |

Sampling Rate = Sample time x respective sensor sample interval

Where:

- Sample time = value in column 2
- CT = $1^{st}$ digit of the value in column 3

- SBE-43 = 2nd digit of value in column 3
- WET Labs = 3rd digit of value in column 3

The science sampling for this file works as follows, looking at the sampling protocol rows:

- First row of sampling protocol:
  - Sampling in depths from 0-50m.
  - Each of the sensors is sampled once every 5 seconds.

    This value is calculated by multiplying the 5 in column 2 by the first digit (1) in column 3 for CT, by the second digit (1) in column 3 for SBE-43F and by the third digit (1) in column 3 for the WET Labs sensor.
  - G&C sampling between 0 and 50m occurs once every 30 seconds.
- Second row of sampling protocol:
  - Sampling depths between 50 and 200m.
  - The CT and WET Labs sensors are sampled once every 5 seconds.

    This value is calculated by multiplying the 5 in column 2 by the first digit in column three for the CT sensor and multiply the 5 in column 2 by the third digit in column three for the WET Labs sensor.
  - The SBE-43F oxygen sensor is sampled once every 10 seconds.

    This value is calculated by multiplying the 5 in column 2 by the second digit in column 3.
  - The G&C sampling occurs once every 60 seconds.
- Third row of sampling protocol
  - Sampling depths between 200 and 300m.
  - The CT sensor is sampled once every 5 seconds.

    This value is calculated by multiplying the 5 in column 2 by the first digit in column 3.
  - The SBE-43F oxygen sensor is not sampled.

    This value is calculated by multiplying the 5 in column 2 by the second digit (0) in column 3.
  - The WET Labs sensor is sampled once every 15 seconds.

    This value is calculated by multiplying the 5 in column 2 by the third digit (3) in column 3.

- The G&C sampling between 200 and 300m occurs once every 120 seconds.

**Note:** The bound on the depth interval is dependent on pressure sensor resolution. Choices made in the science sampling have an impact on the energy use of Seaglider. More samples and shorter G&C intervals use more processor and sensor power than do fewer samples and longer G&C intervals.

### PicoDOS Commands Batch File

This file contains picoDOS direct actions such as resending dive files from Seaglider to the basestation and changing targets out of sequence. Examples of pdoscmds.bat files are below.

If Seaglider is flying toward the target, called WPONE in the example on page 312 under the targets description and the pilot wants to redirect it to head toward WPTHREE at the next surfacing, the pdoscmds.bat file looks like:

target WPTHREE

If the pilot wants Seaglider to resend all files from dive 2 to the basestation, the pdoscmds.bat looks like:

resend_dive 2

See resend_dive on page 312.

Multiple commands can be sent in the same pdoscmds.bat file. For example, if the pilot wants dive 2 resent and wants to change the waypoint Seaglider is heading to, to WPTHREE the pdoscmds.bat file looks like:

target WPTHREE

resend_dive 2

Order of the commands in the pdoscmds.bat file is not critical.

For a full list of commands that can be included in the pdoscmds.bat file, refer to the Appendix C, “Extended PicoDOS® Reference Manual, v66.07” on page 303.

## *Factory-Generated File*

The initial sg_calib_constants.m file is generated at the factory.

### Calibration File

The sg_calib_constants.m file contains calibration information for that particular Seaglider and the sensors installed on it and is used for basestation calculations and MATLAB plots. There should be no need for edits to this file between factory refurbishments. If edits are deemed necessary, do so with caution. An example of a sg_calib_constants.m file can be found in Appendix B, "Seaglider File Formats Manual" on page 275. The values in the sg_calib_constants file for your Seaglider differ from the values listed.

## *Seaglider-Generated Files*

Seaglider generates the following files:

- Capture (.cap)
- Data (.dat)
- Log (.log)

### Capture Files

The capture (.cap) file contains information about all of the actions Seaglider took during the previous dive.

Capture files are a great source of information on Seaglider's performance, especially in error analysis and debugging. However, they can become quite large (multi-hour dives) and are not always sent to the basestation during a surfacing. It is recommended that for the first few dives of the mission, while Seaglider is being tuned for flight efficiency, that the .cap files be sent to the basestation. The parameter **$CAPUPLOAD** (Chapter 5, "Piloting Parameters" on page 83) controls this feature.

The format of the capture file is not as hard and fast as other file formats, but it usually conforms to the following format: time, service, output level, text.

See Appendix B, "Seaglider File Formats Manual" on page 275 for an example of a .cap file.

## Data Files

Seaglider generates the .dat file (an ASCII text file) and transmits it to the basestation for further processing. The first line is the only actual value; all of the following lines are differences. The .dat file serves as the primary conduit for the science data collected by Seaglider. The format is designed to minimize transmission size and, while clear text, is not intended for direct use by users. Each data file covers one dive of information.

The numbers in the data file can be interpreted by the column titles listed in the header line at the top of the file. The first 10 columns of data are the same for each dive and Seaglider. The content of the remaining columns vary depending upon what sensors are installed in that particular Seaglider.

See Appendix B, “Seaglider File Formats Manual” on page 275 for an example of a .dat file.

## Log Files

The .log file serves as a summary record of what happened during the dive. One .log file is made for each dive.

The first portion of the data is a list of Seaglider’s parameters and their values for that dive. See Chapter 5, “Piloting Parameters” on page 83 for an explanation of each parameter.

The second section, beginning with the entry **$GPS1**, contains information concerning the pre-dive period at the surface, what target Seaglider is heading to during the next dive, Kalman filter information, and D_GRID information if bathymetry maps are being used.

The **$GC** labeled lines describe motor actions (pitch, roll, or VBD), one line per motor move. The information listed after the **$GC** lines is data collected at the end of the dive (for example, surface maneuver data, final temperature reading). Some of this data is from the previous surfacing (before the start of the current dive).

The $SM_CC labeled lines contain information on the surface maneuver.

The remaining lines of the log contain information on internal humidity, pressure and temperature, battery power available and amp hours used, ending with a summary of errors encountered during the dive and the GPS position immediately after surfacing.

Not all Seagliders report all of the lines that appear in the example, because the devices installed vary among Seagliders.

### Files Generated by the GPCTD

ALI devices will generate unique output files which can be recognized by the prefix for that sensor, as defined in the .cnf file. For example, the GPCTD generates the following four files per every dive, which use the prefix “pc”:

- pc a.dat
- pc a.eng
- pc b.dat
- pc b.eng

The files labeled with an “a” represent the dive phase while the “b” represents the climb phase.

## *Basestation-Generated Files*

The basestation generates the following files, using data provided by Seaglider:

- ASCII (.asc)
- Baselog (baselogXXXXXXYYYYYY.com)
- Binned profiles (.bpo)
- Communications (comm.log)
- Engineering (.eng)
- netCDF (.nc)
- Pagers (.pagers)
- .pro
- Processed files cache (processed_files.cache)
- Private (.pvt)
- URLs (.urls)

### ASCII Files (.asc)

The .asc, or ASCII, files are created on the basestation. They are the reconstituted (uncompressed, reassembled, and differentially summed) versions of the data (.dat) files created on Seaglider. See Appendix B, “Seaglider File Formats Manual” on

page 275 for a description of the column names. The entry NaN indicates that there was no sample returned for that sensor. Either the sensor was not installed, or the sensor was not enabled for that sample/deployment, as controlled by the science file.

### Baselog File

The baselog file is produced by the basestation, and logs the output from the scripts that perform the conversion and notification functions of the basestation. It is written during each invocation.

This file is the first place to look when debugging problems with the data conversion. If the basestation cannot process a file, it sends an alert to any contact listed in the .pagers file who requested alerts. See “Pagers File” on page 250 for more information.

The naming convention for the baselogXXXXXXYYYYYY.com file is:

- XXXXXX — the date in ddmmyy format
- YYYYYY — the time in hhmmss in the timezone on the basestation

### Binned Profiles Files

The binned profiles (.bpo) files contain the same data as the .pro files except that .bpo files are binned or averaged into depth intervals specified by the user.

### Communications File

The communications (comm.log) file is resident in Seaglider’s home directory on the basestation. The plain text file is appended to during each communication session with Seaglider. So, not only is it a complete record of Seaglider’s communications over an entire deployment, it can become quite large.

To monitor the end of the file for any new additions, type **tail –f comm.log** on the command line in Seaglider’s home directory.

See Appendix B, “Seaglider File Formats Manual” on page 275 for an example of a comm.log file.

### Engineering Files

The .eng, or engineering files are created on the basestation. They restate the data contained in the .asc file, but converted into engineering units. The column titles are described in Appendix B, “Seaglider File Formats Manual” on page 275.

The first 11 columns of data are the same for each dive and Seaglider. The content of the remaining columns vary depending upon what sensors are installed in that particular Seaglider.

### netCDF Files

The netCDF (.nc) file captures all processed files and is self-documenting. Read-write access to netCDF is provided by the software libraries supplied by UCAR (University Corporation for Atmospheric Research). The netCDF file is meant primarily for sharing data between scientific users.

## Pagers File

The .pagers file controls the automatic email, pager, and text message notification system. The file allows any of four types of messages (gps, alerts, recov, comp) to be sent to any valid email address or phone number. This service is run by the data conversion script, which is invoked by a Seaglider logout or disconnection.

Lines beginning with a # are comment lines and are ignored in processing.

The gps message is sent after every dive and is the most recent GPS position. The alerts message is sent when the basestation has a problem converting a file or files.

The file may be corrupt and have to be resent by Seaglider to the basestation using the pdoscmds.bat file (see “PicoDOS Commands Batch File” on page 245 for details).

The recov message is sent when Seaglider goes into recovery mode. This message includes the most recent GPS position and the recovery mode.

The comp message is sent when processing of the lastest files received from Seaglider is completed by the basestation.

An example .pagers file is shown in Appendix B, “Seaglider File Formats Manual” on page 275.

### .pro Files

The .pro files contain the time stamp and scientific data that was acquired during the dive, such as temperature and salinity, optical and dissolved oxygen data.

See Appendix B, “Seaglider File Formats Manual” on page 275 for an example of the column headers in this file.

### Processed Files Cache File

The processed files cache (processed_files.cache) contains the list of files that have been processed by the basestation and the time of processing. To force a file to be re-processed, delete the corresponding file from this file. Comment lines are preceded by a #.

### Private Files

The .pvt, or private, files are created on the basestation. They contain information that was originally in the .log file that could pose a security problem if propagated off of the basestation (that is, basestation telephone numbers and passwords). Thus, the data is stripped from the .log file and placed in the matched .pvt file.

### URLs File

The .urls file is read by the basestation, following processing of dive data (triggered by a Seaglider logout). It specifies URLs on which to run GET for each processed dive. This can be used for any supported http: function, and is mainly used to poll for data transfers to support visualization servers.

The first entry on the line is the timeout (in seconds) to wait for a response to the GET. It is separated from the URL by a tab. convert.pl adds arguments "instrument_name=sg& dive=" with the proper separator.

Comments in the file are indicated by a #.

It is up to the user to set up the .urls file as directory structure and location of data visualization software differs from user to user.

### Mailers

The .mailers file is used by the basestation to e-mail generated files. Each line of this file contains an e-mail address followed by a comma separated list of options. The grammar for this line is specified as

smtp_mail_address[,body|msgperfile|kkyy_subject|gzip] [,[eng| log| pro| bpo| csv| asc| cap| comm| dn_kkyy| up_kkyy| nc| mission_ts| mission_pro]+] | ,all

These options are as follows:

- body - send the files in the mbody of the message and not as attachments, incompatible with the gzip, nc, mission_ts and mission_pro options

- msgperfile - Only send one message per file
- kkyy_subject - Use Navy specified XBTDATA as the subject line instead of informative subject
- gzip - Compress attachments using GZIP
- eng - Send .eng files
- log - Send .log files
- pro - Send .pro files
- bpo - Send .bpo files
- csv - Send .csv files
- asc - Send .asc files
- cap - Send .cap files
- comm - Send comm.log files
- dn_kkyy - Send downcast kkyy files
- up_kkyy - Send upcast kkyy files
- nc - Send compressed NetCDF files
- mission_ts - Send compressed mission_ts files
- mission_pro - transmit compressed mission profile data
- all - send all files listed above

Only files that have been generated will be transmitted. For instance if NetCDF or Dive profile data has not been configured to be generated there will be no files to send even if the .nc and .pro options are chosen.

An example line can be found below:

kkyyuser@billybob.com,msgperfile,body,kkyy_subject,dn_kkyy,up_kkyy

# *Files Stored on Seaglider*

The following files are stored on Seaglider.

### Bathymetry Map Files

Map files provide Seaglider with geographic environmental information, specifically bathymetry about a given region of the ocean.

Seaglider can carry up to 999 bathymetry maps (the files are named bathymap.nnn, where .nnn is the map number), but in practice far fewer are typically on board. These maps are not required for Seagliders to fly, but provide a low energy means to determine apogee depth.

Note that the values in the bathymap files represent the depth at which Seaglider will start its apogee maneuver, NOT the actual bottom depth.

For additional information on bathymetry maps see Chapter 6, “Pre-Deployment Tasks” on page 159.

### Battery File

The Battery file is used by Seaglider to keep track of power consumption throughout the time Seaglider is using the battery pack. This file is NOT intended to be edited by the user.

### Capvec File

The Capvec file is parsed by Seaglider and updates one or more elements of the Capture Vector. Normally, this file is not used except for Seaglider provisioning. See the capvec and parse_capvecfile commands in Appendix C, “Extended PicoDOS® Reference Manual, v66.07” on page 303 for details on updating the Capture Vector.

The capvec file format is line oriented. Lines can be comment lines, in which case the first character must be a /. All other lines are updates to the Capture Vector and are documented under the capvec command in the Appendix C, “Extended PicoDOS® Reference Manual, v66.07” on page 303.

### Compass Calibration File

The Compass Calibration file is generated when the compass is calibrated inside the assembled Seaglider at the factory. The calibration corrects for the effects of the metal on the compass readings. This file is NOT intended to be edited by the user.

CHAPTER 10

# *Recovery and Disassembly*

The following topics are covered:


- "Recovery Loop" on page 255
- "Recovery Phase" on page 256


## *Recovery Loop*

In recovery, Seaglider enters a loop of obtaining a GPS fix and communicating with the basestation **$T_RSLEEP** minutes. In practice, there are about two minutes of overhead in this process, so that the actual time between phone calls is closer to **$T_RSLEEP** + 2 minutes. This recovery loop can be exited by sending a **$RESUME** directive to Seaglider in the cmdfile. Seaglider then continues diving.

**Note:** The following recovery procedures were developed by the University of Washington and adopted by iRobot as a method for recovering Seaglider from a small vessel. Seaglider missions and/or weather conditions can dictate a modified or even completely different method of recovery.

## *Recovery Phase*

The recovery phase is entered either by the $QUIT command issued by the pilot via the cmdfile (to keep Seaglider at the surface) or by an error condition detected by Seaglider operating software. In recovery phase, Seaglider stays on the surface and acquires a series of GPS fixes which are sent to the basestation so that Seaglider can be recovered.

To begin recovery, the field team must in the vicinity of the anticipated surfacing location of Seaglider, with all support equipment, receiving Seaglider's most recent GPS position via text message or voice contact with the pilot.

### Field team:

When the field team arrives in the recovery area, they should instruct the pilot to give Seaglider the **$QUIT** command so Seaglider does not dive again and stays on the surface.

### Pilot:

Give Seaglider the **$QUIT** command via the cmdfile.

### Field team:

Complete the following steps:

1. Transit to the last set of GPS coordinates.
2. Search for Seaglider by looking for the antenna sticking out of the water.

   Binoculars may be helpful. Use the acoustic recovery system to ping Seaglider if visual conditions are poor. Follow the instructions in Chapter 6, "Pre-Deployment Tasks" on page 159 to use the transducer.
3. Once a visual of Seaglider has been made:
   a. Transit to Seaglider.
   b. Make sure one of the handles of the cradle is tied off to the boat.
   c. Grab Seaglider's antenna mast as close to the rear of the aft fairing as possible.
   d. Put the cradle in the water alongside Seaglider.
   e. Maneuver the cradle between the boat and Seaglider.
   f. Pull Seaglider up and put the nose into the cradle.

g. Hold on to both Seaglider and the cradle and pull both up out of the water just far enough so that the water can drain out of the nose of Seaglider.

h. When Seaglider has finished draining, pull Seaglider and the cradle the rest of the way out of the water.

i. Connect at least one of the safety straps on the cradle around Seaglider.

j. In a two-person lift, put the glide/cradle in a safe place on the deck of the boat.

k. Secure Seaglider to the boat.

l. Wand Seaglider off.

m. Since the field laptop is not connected to Seaglider, have the pilot continue to monitor Seaglider for more phone calls to the basestation. If Seaglider is calling the basestation, it is not powered down and needs to be wanded off again.

4. Replace sensor dust caps.

5. Disassemble Seaglider – rudder, antenna and wings, in that order following the directions below. If sea conditions are poor, disassembly may be better accomplished back on shore.

   a. Rudder

      i Slide Seaglider backwards until the rectangular slot on the aft end fairing is clear of the cradle end (Figure 3-7 on page 45).

      ii Remove the screws holding the rudder in place.

      iii Remove the rudder.

      iv Set aside the rudder for packing and put the screws back in the spare hardware kit.

   b. Antenna

      i Slide the antenna mast out of the aft end of the fairing. Ease the cables out while doing this but do not disconnect them.

      ii Fold the antenna around so that it can be secured in the cradle.

   c. Wings

      i Doing one wing at a time, remove the upper screws holding the wing in place then rotate Seaglider slightly and remove the lower screws. Repeat for the other wing.

      ii Set the wings aside for packing and put the screws back in the spare parts kit.

6. Wash Seaglider following the instructions in Chapter 12, “1KA Seaglider Refurbishment” on page 263.

7. Pack Seaglider in its shipping crate (see Figure 2-1 on page 30).
8. Wash the Launch and Recovery Cradle.
9. Disassemble the Launch and Recovery Cradle
   a. Remove the wing nuts and eye bolts from the small holes in the end plates with carrying handles. Put the hardware in the plastic spares kit.
   b. Detach the two upper rails from the end plates with carrying handles by removing the large bolts using the ¾” wrench. Put the hardware in the plastic spares kit.
   c. Slide the straps off of the upper rails
   d. Slide the yellow mesh from the upper rails and roll up.
   e. Detach the two lower rails from the end plates with carrying handles by removing the large bolts using the ¾” wrench. Put the hardware in the plastic spares kit.
   f. Pack the Launch and Recovery Cradle parts in the shipping crate (see Figure 2-1 on page 30).

CHAPTER 11

# *Operator Level Maintenance*

The following topics are covered:


- "Cleaning Seaglider" on page 259
- "Deep Cleaning Seaglider" on page 261


## *Cleaning Seaglider*

Cleaning Seaglider is not a scheduled task, but is done on an as-required basis, such as after a mission. After recovery from a salt water mission, Seaglider should be rinsed as described below to prevent salt buildup.

### Tools Required:

- Low-pressure water hose
- Clean cloth

### Consumables:

- Deionized water

**Parts:**
None

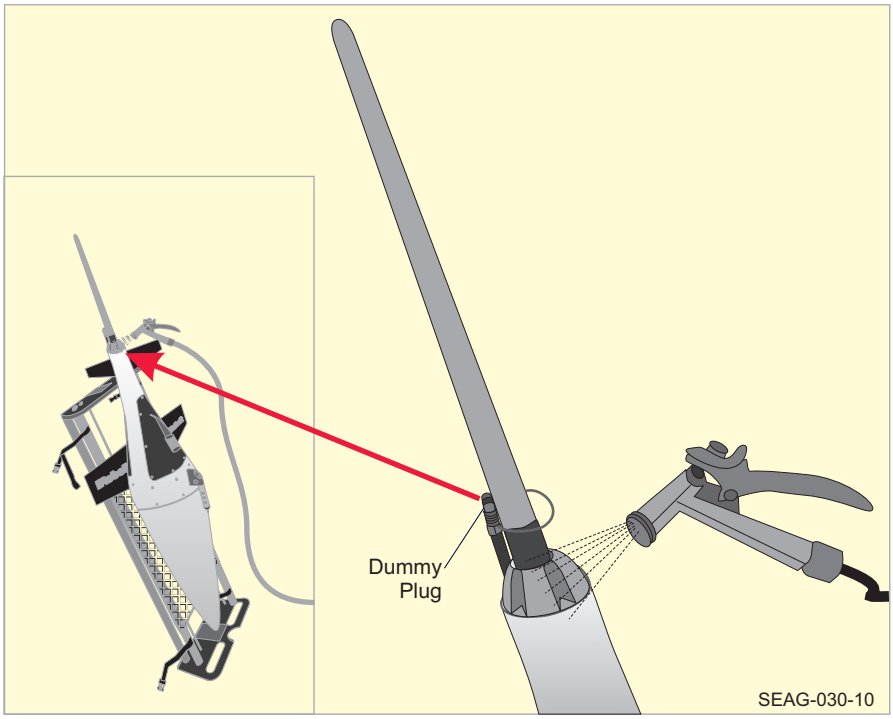
To clean Seaglider:

1. Make sure the vehicle is powered down.
2. Remove the dust caps from the science sensors.
3. For cleaning of the CTD sensor, Sea Bird recommends flushing the conductivity cell with a dilute bleach solution to eliminate growth of bio-organisms, and eliminating the use of acid in most cases.
4. Wash down the exterior of the Seaglider allowing the water to flow through the conductivity and temperature sensor.

**Caution:** DO NOT use a power washer near the sensors.

5. Place the Seaglider/cradle combination so that Seaglider's nose is downward.
6. Flush the conductivity and temperature sensor with deionized water.
7. If your Seaglider is configured with additional sensors, be sure to rinse thoroughly with deionized water.

**FIGURE 11-1. Post Deployment Seaglider Cleaning**




8. Flush the CTD sensor with deionized water.

   The conductivity cell should be left moist but not filled with water. Should the water freeze inside the cell the glass tubing could crack.

9. Replace the dust caps on all of the science sensors.

## *Deep Cleaning Seaglider*

If there is more than one week between missions, if Seaglider has been deployed for multiple months or if there are signs of biofouling on the fairing, you must deep clean Seaglider by completing the following steps:

1. Remove the wings, rudder, fore and aft fairing so that only the antenna remains attached to the pressure hull.

2. Using a fresh water tank, soak all pieces overnight (12+ hours).
3. Dry with a soft cloth.
4. Put Seaglider back in shipping case, if appropriate.

CHAPTER 12

# *1KA Seaglider Refurbishment*

The following topics are covered:


- "iRobot Refurbishment" on page 263


## *iRobot Refurbishment*

A Seaglider refurbishment is necessary after the lithium primary battery pack(s) have been depleted or there is not enough battery power remaining to complete the next mission. Replacement can only be performed by the original equipment manufacturer (OEM) trained personnel at iRobot.

To return your iRobot maritime product for refurbishment, calibration or other service, please provide the information below, so we can serve you better and prevent delays in the return of the product and/or instruments:

1. Request a quote/service request for refurbishment, calibration or repair from iRobot through your sales contact or e-mail directly to maritimesupport@irobot.com
2. Once you receive a quote/service request please review for accuracy and approve the quote in writing to maritimesupport@irobot.com

3. Provide a purchase order or other approved method of payment reflecting amount of quote/estimate provided
4. You will receive a Service Request (SR) number from iRobot including shipping instructions. Do not ship the items until you have approved the quote/estimate from iRobot contracts department and have issued a purchase order.
5. Reference Chapter 12 1KA Seaglider Refurbishment in the User Guide for additional information.
6. E-mail additional questions to MaritimeSupport@iRobot.com

If returning an entire Seaglider unit for service, please perform a one Self Test and 1 Simulated Dive utilizing the internal battery power of the Seaglider and send the following files to maritimesupport@irobot.com prior to shipment of the unit for service:

| Self-Test Files: | Simulated Dive Files: |
|---|---|
| ptxxxxxxx.cap | pxxxxxxx.cap |
| ptxxxxxxx.log | pxxxxxxx.log |
| ptxxxxxxx.eng | pxxxxxxx.eng |
| ptxxxxxxx.pvt | pxxxxxxx.pvt |
| sg_calib_constants.m | sg_calib_constants.m |
| pdoscmds.bat | pdoscmds.bat |
| targets | targets |
| science | science |
| cmdfile | cmdfile |

Also, please provide the following items prior to shipment of your iRobot maritime product for service:

- Latest version of the Seaglider's Trim Sheet
- List of parameters that have changed from the factory default values
- List desired services, spares, and/or diagnostics for your Seaglider product (see tables below)
- List of sensor serial #'s to be recalibrated (if ordering OEM sensor recalibration service):

| Sensor Type | Serial # |
|---|---|
| Choose an item. | |
| Choose an item. | |
| Choose an item. | |
| Choose an item. | |
| Choose an item. | |

## *Contact Information*

| | |
|---|---|
| **Your Name:** | |
| **Institution/Company:** | |
| **Shipping/Delivery Address:** | |
| **Phone #:** | |
| **Fax #:** | |
| **E-mail Address:** | |

## *Service Information*

| | |
|---|---|
| **Desired Ship Date:** | |
| **Unit Quantity:** | |
| **Unit Serial Number(s):** | |
| **Desired Return Date:** | |

## *Standard Services*

| Service | Description | Part Number | Quantity |
|---|---|---|---|
| Refurbishment Service (ElectroChem 24V batteries) | Includes the factory-based replacement of lithium primary batteries, inspection of internal components and re-ballasting. Requires shipping Seaglider to iRobot. | 4255726 | |
| Refurbishment Service (Saft 24V batteries) | Includes the factory-based replacement of lithium primary batteries, inspection of internal components and re-ballasting. Requires shipping Seaglider to iRobot. | 4249888 | |
| OEM Sensor Recalibration | Recalibration of Seaglider sensors at OEM. Service is ordered separately for each sensor. Includes re-installation and testing. | 4249889 | |
| Ocean Re-Trim and Ballast | Re-trim and re-ballast of Seaglider after refurbishment to enable faster re-deployment. Process requires an ocean launch, piloting, and recovery prior to delivery to the customer. | 4255839 | |

## *Upgrade Services*

| Service | Description | Part Number | Quantity |
|---|---|---|---|
| Enhanced Buoyancy Engine | The Enhanced Buoyancy Engine supports single-pump operation in water depths shallower than 120 meters resulting in a significant reduction in power consumption. Does not include ocean re-trim and ballast. Requires shipping Seaglider to iRobot. | 4270717 | |
| Upgrade to Seaglider 1KA | Upgrades a Seaglider purchased from University of Washington (before July 2009) to the current configuration of the iRobot 1KA Seaglider. Includes an updated mass shifter, an ElectroChem 24V lithium primary battery, and a 10V lithium primary battery. Does not include an upgrade to Ogive fairings. | 4261537 | |
| Sea-Bird Electronics GPCTD | Upgrade from a Sea-Bird CT sail to a Sea-Bird GPCTD. Replaces older CT sail with a Sea-Bird Conductivity, Temperature, and Depth sensor. Requires Ogive Fairings Upgrade. | 4292639 | |
| Ogive Fairings | Replacement of fairings to the larger capacity Ogive fairings, allowing for greater sensor payload capacity and improved endurance. Mandatory with Sea-Bird GPCTD upgrade. | 4339550 | |

## *Replacement Spares, Sensors & Accessories*

| Part | Part Number | Quantity |
|---|---|---|
| Standard Antenna Mast Kit | 4199332-00001<br>4199332-00002 | |
| Short Antenna Mast Kit | 4277576-00001<br>4277576-00002 | |
| Forward Fairing Kit | 4249893<br>4294322-00001 | |
| Aft Fairing Kit | 4249894<br>4294323-00001 | |
| Nose Kit (Ogive) | 4327116 | |
| Wings Kit | 4249895<br>4326606 | |
| Rudder Kit | 4249896<br>4326607 | |
| Photosynthetically Active Radiation (PAR)<br>Biospherical Instruments Sensor QSP-2150 | 4287926 | |
| Dissolved Oxygen Sensor: Aanderaa 4330/4330F | 4196239 | |
| Dissolved Oxygen Sensor: Sea-Bird SBE 43F<br>(Free Flow) | 4192455 | |
| Dissolved Oxygen Sensor: Sea-Bird SBE 43F<br>(Pumped) | 4289668 | |
| Backscatter Meter/Fluorometer<br>WETLabs BB2FL-VMT 470/700/CHL-A | 4249861 | |
| Backscatter Meter/Fluorometer<br>WETLabs BB2FL-VMT 532/880/CHL-A | 4314144 | |
| Backscatter Meter/Fluorometer<br>WETLabs BB2FL-VMT 532/CDOM/CHL-A | 4273157 | |

| Part | Part Number | Quantity |
|---|---|---|
| Backscatter Meter/Fluorometer<br>WETLabs BB2FL-VMT 650/CDOM/CHL-A | 4192457 | |
| Sea-Bird GPCTD | 4292639 | |
| Sea-Bird CT Sail | 4249882 | |
| Shore Station System | 4196238 | |
| Communication Cable | 4196985 | |
| Collapsible Field Cradle | 4320926 | |
| Magnetic Power Wand | 4199416 | |
| Hardware Field Kit | 4199502 | |
| Hardened Shipping Case | 4311445 | |
| Argos Mount Kit (Standard Antenna) | 4266416 | |
| Argos Mount Kit (Short Antenna) | 4327337 | |
| Communications Cable with External Power Adapter | 4196986 | |

## *Diagnostics*

Please list any issues you have encountered with your Seaglider product and/or specific functions you would like iRobot to investigate during your refurbishment, calibration, or other service:

| Issue/Function for Diagnosis | Description | Quantity |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

**Please be advised that lithium battery shipments are controlled by the Department of Transportation (DOT), International Civil Aviation Organization (ICAO), and the International Air Transport Association (IATA). Under the US DOT regulations, please review requirements under 49 CFR 172.101 and Special Provisions 29, 188, 189, 190, A54, A55, A101, and A104 and packing instruction 49 CFR 173.185. Under the IATA regulations,**

**please review 4.2 List of Dangerous Goods and Special Provision A48, A88, A99, A154, A164 and Packing Instruction 968, 969, or 970.**

APPENDIX A

# *System Specifications*

**TABLE A-1. 1KA Seaglider Specifications**

| Feature | Specification |
|---|---|
| Body Size | 1.8-2.0 meters long (configuration dependent),<br>30 cm max. diameter |
| Wing Span | 1 m |
| Antenna mast length | Between .43 m and 1 m (configuration dependent) |
| Weight | 52 kg (dry) |
| Batteries | Lithium Sulfuryl Chloride Primary Batteries, 24V and 10V packs, 17 MJ |
| Battery Endurance (fully charged batteries) | • Up to 10 months (mission dependent) |
| Computer Software | • Complete data transmitted after every dive<br>• Web-based information interface<br>• Control and system commands can be transmitted before each dive |
| RF Data Telemetry | Iridium satellite data telemetry |

**TABLE A-2. 1KA Seaglider performance specifications**

| Feature | Specification |
|---|---|
| Operating Depth Range | 20 to 1,000 m (configuration dependent) |
| Maximum Travel Range/Duration | 4,600 km (650 dives to 1-km depth) |
| Typical Speed | 25 cm/s (1/2 knot) |
| Glide Angle | $16^{o}$ to $45^{o}$ (1:3.5 to 1:1 slope) |

**TABLE A-3. Standard Sensors**

| |
|---|
| Paine Pressure Sensor |

**TABLE A-4. Optional Sensors**

| |
|---|
| Sea-Bird 43F dissolved oxygen sensor (pumped) |
| Sea-Bird dissolved oxygen sensor (unpumped) |
| Sea-Bird CT Sail |
| Aanderaa dissolved oxygen |
| WET Labs ECO Pucks™ |
| Photosynthetically Active Radiation (PAR) sensor |
| Payload Conductivity Temperature Density (GPCTD) sensor |

**TABLE A-5. Mechanical Features**

| |
|---|
| Isopycnal pressure hull |
| No external moving parts |
| Low drag, flooded fairing |
| Glider Payload Conductivity Temperature Density (GPCTD) sensor |

**TABLE A-6. Electrical Features**

| |
|---|
| Ultra-lower power micro-processor |
| High-capacity compact-FLASH memory |
| 4 open serial channels for sensors |
| 1 open frequency channel for sensors |

**TABLE A-7. Guidance and Control (G&C)**

| |
|---|
| Dead reckoning between surface GPS fixes using 3-axis digital compass |
| Kalman filter prediction for mean and oscillatory currents |
| Acoustic altimetry systems for near-bottom dives |
| Bathymetry map system for low-energy dives |

**TABLE A-8. Operational Modes**

| Mode | Description |
|---|---|
| Transect survey | Profiles using a sequence of waypoints |
| Virtual mooring | Continuously profiles at a single location |
| Surface drift | Stays on the surface with the antenna up for GPS and data telemetry |
| Bottom loiter | Loiters at designated depth for a specified amount of time before surfacing |
| Sub-surface porpoise | Profiles a specified depth range without surfacing |

APPENDIX B

# *Seaglider File Formats Manual*

SCHOOL OF OCEANOGRAPHY
and
APPLIED PHYSICS LABORATORY
UNIVERSITY OF WASHINGTON
Version 66.07 December 2009

## *Chapter 1*

### Conventions and Introduction

#### *1.1 Conventions*

Example files are given in bold Courier font. Direct annotations of files are given in smaller font. Parameters are in UPPER CASE BOLD font, and have a preceding $. File names that are used in Seaglider command, control, or operations are given in lowercase bold font. Documents and sections of documents are italicized.

123 is used throughout this document as a placeholder for Seaglider serial number, and 55 is used as a placeholder for dive number. Many file names include a three digit Seaglider serial number, followed by a four digit dive number, both with preceding zeros (e.g. p1230055.log). Numerals after the dot in a file name are

represented by 0's and, when additional numerals are needed, 9's. Because they represent various meanings, numerals after the dot are always annotated the first time the file name appears, and in the file description heading.

### *1.2 Introduction*

This manual is designed to help the Seaglider user identify and interpret files he or she will encounter on the basestation. It is to be used in conjunction with the Seaglider Pilot's Guide, Parameter Reference Manual, and Extended PicoDOS Reference Manual.

#### *1.2.1 List of Files Found on the Basestation*

(using SG132, dive 55, for example file names)

| | | |
|---|---|---|
| | **processed_files.cache**<br>**baselog_080221110101**<br>**baselog.log**<br>**sg_calib_constants.m**<br>**cmdfile**<br>**comm.log** | |
| These files are described in the document below. | **p1230055.asc**<br>**p1230055.cap**<br>**p1230055.dat**<br>**p1230055.eng**<br>**p1230055.log**<br>**p1230055.pro**<br>**p1230055.bpo**<br>**p1230055.pvt** | p indicates that these files have been processed by the basestation. They are the files that contain information from the glider, for use by the pilot, operator, and scientist. |
| | **p1230000.prm** | This file is sent at the end of a self test. Contains a list of the parameters and their settings at the time of the self test, and some information about the transmission of files during the self test. |
| | **cmdedit.log**<br>**targedit.log**<br>**sciedit.log** | These files are made by the basestation, and document each change made to the command file, targets file, and science file using cmdedit, targedit, and sciedit. |
| | **comm_merged.log** | Merged comm log and history |
| | **history.log** | Record of shell commands |
| | **cmdfile.0**<br>**targets.0**<br>**science.0**<br>**p1230055.000.pdos** | Every time a cmdfile, targets file, or science file is taken up by the glider, it is saved on the basestation and renamed to include the dive number. PDOS command files are also saved, but already include the dive number, so they are saved with a serial number. If there are multiple calls on one surfacing, a cmdfile is sent each time, and a serial number is added after the dive number. |

| | | |
|---|---|---|
| These files are intermediates found on the basestation. They are used to create the processed files documented in this manual. Characters in the file names indicate the following:<br>st : The file is from a self-test. If from a normal dive, this prefix will be pt<br>b: has had duplicate sections removed "Bogue Syndrome processing"<br>1a: has been stripped of the padding characters added for transmission from the Seaglider.<br>u: uncompressed<br>z: zipped<br>r: raw; a reconstruction of the raw ASCII text file on the glider<br>x: The following sequence number is in the hexidecimal system | **st0055du.1a.x00**<br>**st0055du.r**<br>**st0055du.x00** | d indicates that these intermediate files will be used to create a data file. |
| | **st0055lu.1a.x00**<br>**st0055lu.x00** | l indicates that these intermediate files will be used to create a log file. |
| | **st0055kz.1a.x02**<br>**st0055kz.1a.x03s**<br>**t0055kz.b.1a.x04**<br>**st0055kz.b.x04**<br>**st0055kz.r**<br>**st0055kz.x00**<br>**st0055kz.x01** | k indicates that these intermediate files will be used to create a capture file. |
| | **st0055kz.x00.PARTIAL.1** | Partial files appear when the basestation does not receive a complete file from the Seaglider, and is unable to process it. Transmission errors are addressed in the Communications Log section of this document, and in the Seaglider Pilot's Guide. |

### *1.2.2 Data Flow Map*

# *Chapter 2*

## File Descriptions

This section describes the files relevant to the Seaglider user. Where appropriate, excerpts from real files, with explanatory annotation, are shown.

## 2.1 Processed Files

### *2.1.1 Log File*

(p1230055.log)

One log file is made for each dive. The first portion of the data is a list of the Seaglider's parameters and their values for that dive. See the Parameter Reference Manual for more information. The second section, beginning with the entry $GPS1, contains information concerning the pre-dive period at the surface. The $GC-labeled lines describe motor actions (pitch, roll, or VBD), one line per motor move. The information listed after the $GC lines are data collected at the end of the dive (surface maneuver data, final temperature reading, etc). Some of this data is from the previous surfacing (before the start of the current dive). Not all Seagliders will report all of the lines that appear in the example given here, because the devices installed vary among Seagliders.

### Example Log File

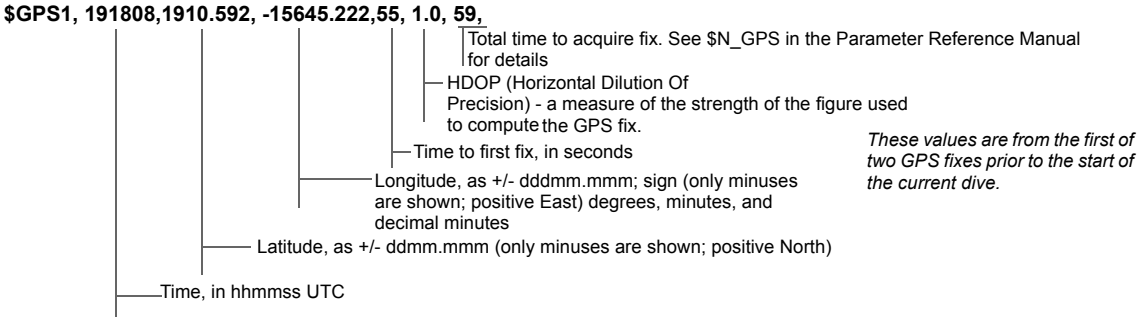
```
 version: 66.06 Seaglider operating code
 glider: 123 Seaglider serial number
 mission: 1 counter settable by pilot or launch operator
 dive: 055 dive number
 start: 7 17 106 19 24 20, day and time (UTC) of start of dive
                          second (UTC, starting with 0)
                       minute (UTC, starting with 0)
                    hour (UTC, starting with 0)
                 year after 1900
              day of month
           month
data:
$ID,123
$MISSION,1
$DIVE,55
$D_SURF,2
$D_FLARE,3
$D_TGT,990
$D_ABORT,1090
$D_NO_BLEED,500
$D_FINISH,0
$T_DIVE,220
$T_MISSION,275
$T_ABORT,1440
$T_TURN,225
$T_TURN_SAMPINT,5
$T_NO_W,120
$USE_BATHY,0
$USE_ICE,-1
$D_OFFGRID,1001
$T_WATCHDOG,10
$RELAUNCH,1
$APOGEE_PITCH,-5
```

$MAX_BUOY,225
$COURSE_BIAS,0
$GLIDE_SLOPE,30
$SPEED_FACTOR,1
$RHO,1.0275
$MASS,52202
$NAV_MODE,0
$FERRY_MAX,60
$KALMAN_USE,1
$HD_A,0.003
$HD_B,0.0099999998
$HD_C,9.9999997e-06
$HEADING,-1
$ESCAPE_HEADING,0
$ESCAPE_HEADING_DELTA,10
$TGT_DEFAULT_LAT,21
$TGT_DEFAULT_LON,-158.3
$TGT_AUTO_DEFAULT,0
$SM_CC,400
$N_FILEKB,4
$FILEMGR,0
$CALL_NDIVES,1
$COMM_SEQ,0
$N_NOCOMM,1
$N_NOSURFACE,0
$PITCH_MIN,331
$PITCH_MAX,3664
$C_PITCH,2720
$PITCH_DBAND,0.1
$PITCH_ADJ_DBAND,0.5
$PITCH_ADJ_GAIN,0.03
$PITCH_MAXERRORS,1
$ROLL_DEG,45
$ROLL_MAX,4000
$ROLL_MIN,120
$PITCH_CNV,0.0046000001
$P_OVSHOOT,0.039999999
$PITCH_GAIN,16
$PITCH_TIMEOUT,20
$PITCH_AD_RATE,150
$UPLOAD_DIVES_MAX,-1

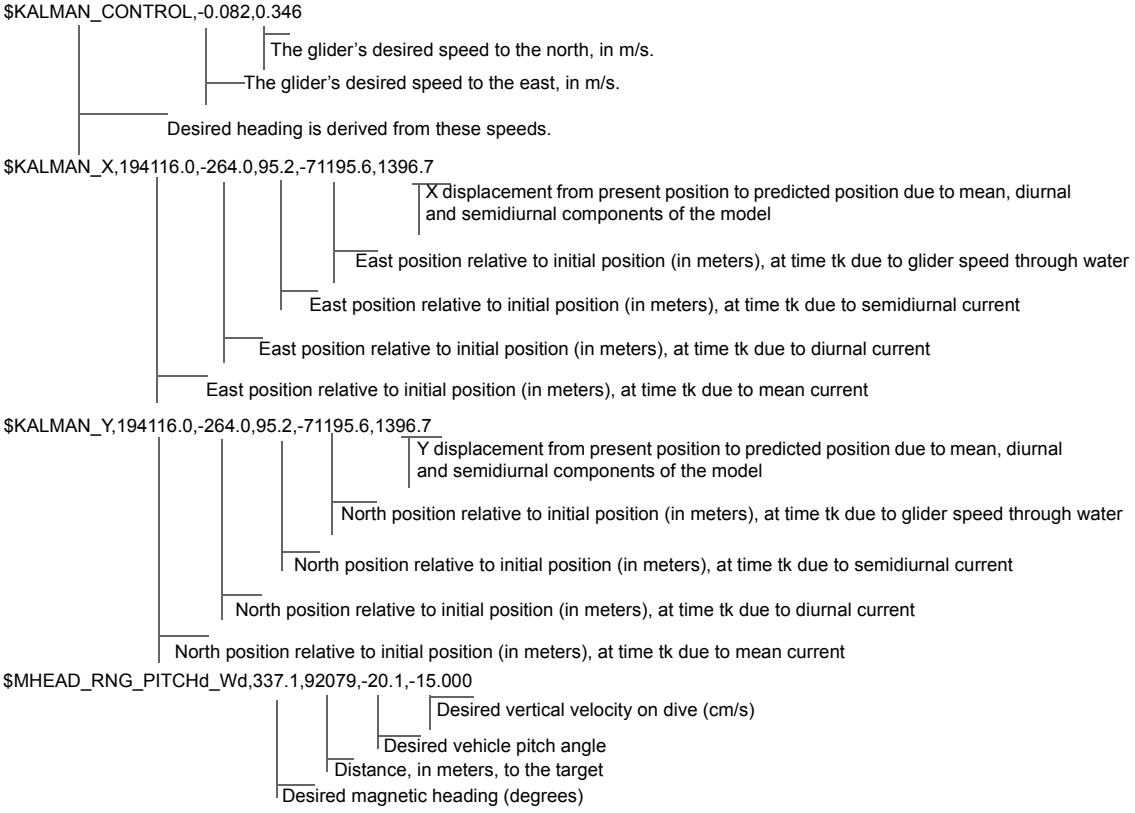
$CALL_TRIES,5
$CALL_WAIT,60
$CAPUPLOAD,0
$CAPMAXSIZE,100000
$T_GPS,15
$N_GPS,20
$T_GPS_ALMANAC,0
$T_GPS_CHARGE,-47579.566
$T_RSLEEP,3
$C_ROLL_DIVE,2150
$C_ROLL_CLIMB,2225
$HEAD_ERRBAND,10
$ROLL_CNV,0.028270001
$ROLL_TIMEOUT,15
$R_PORT_OVSHOOT,62
$R_STBD_OVSHOOT,42
$ROLL_AD_RATE,500
$ROLL_MAXERRORS,0
$ROLL_ADJ_GAIN,0
$ROLL_ADJ_DBAND,0
$VBD_MIN,704
$VBD_MAX,3940
$C_VBD,2956
$VBD_DBAND,2
$VBD_CNV,-0.24529999
$VBD_TIMEOUT,720
$PITCH_VBD_SHIFT,0.0020000001
$VBD_PUMP_AD_RATE_SURFACE,5
$VBD_PUMP_AD_RATE_APOGEE,4
$VBD_BLEED_AD_RATE,8
$UNCOM_BLEED,20
$VBD_MAXERRORS,1
$CF8_MAXERRORS,0
$AH0_24V,91.800003
$AH0_10V,61.200001
$MINV_24V,19
$MINV_10V,8
$FG_AHR_10V,6.94801 $FG_AHR_24V,6.73398 $PHONE_SUPPLY,2
$PRESSURE_YINT,-9.1756201
$PRESSURE_SLOPE,9.1530041e-05
$AD7714Ch0Gain,64

See Parameter Reference Manual for information on parameters reported in the log file.

$TCM_PITCH_OFFSET,0
$TCM_ROLL_OFFSET,0
$ALTIM_BOTTOM_PING_RANGE,0
$ALTIM_TOP_PING_RANGE,0
$ALTIM_BOTTOM_TURN_MARGIN,0
$ALTIM_TOP_TURN_MARGIN,0
$ALTIM_TOP_MIN_OBSTACLE,1
$ALTIM_PING_DEPTH,0
$ALTIM_PING_DELTA,0
$ALTIM_FREQUENCY,13
$ALTIM_PULSE,2
$ALTIM_SENSITIVITY,4
$XPDR_VALID,0
$XPDR_INHIBIT,90
$INT_PRESSURE_SLOPE,0.0097660003
$INT_PRESSURE_YINT,0
$MOTHERBOARD,4
$DEVICE1,2
$DEVICE2,20
$DEVICE3,37
$DEVICE4,-1
$DEVICE5,-1
$DEVICE6,-1
$COMPASS_DEVICE,33
$PHONE_DEVICE,48
$GPS_DEVICE,32
$RAFOS_DEVICE,-1
$XPDR_DEVICE,24
$SIM_W,0
$SIM_PITCH,0
$SEABIRD_T_G,0.004327164
$SEABIRD_T_H,0.00064159534
$SEABIRD_T_I,2.4326842e-05
$SEABIRD_T_J,2.4823044e-06
$SEABIRD_C_G,-10.256908
$SEABIRD_C_H,1.181479
$SEABIRD_C_I,-0.0036624616
$SEABIRD_C_J,0.00030102869


$GPS1, 191808,1910.592, -15645.222,55, 1.0, 59,
Total time to acquire fix. See $N_GPS in the Parameter Reference Manual for details
HDOP (Horizontal Dilution Of Precision) - a measure of the strength of the figure used to compute the GPS fix.
Time to first fix, in seconds
Longitude, as +/- dddmm.mmm; sign (only minuses are shown; positive East) degrees, minutes, and decimal minutes
Latitude, as +/- ddmm.mmm (only minuses are shown; positive North)
Time, in hhmmss UTC
These values are from the first of two GPS fixes prior to the start of the current dive.

| | |
|---|---|
| $_CALLS,1 | Total number of calls that were made in an attempt to connect on the previous surfacing. |
| _XMS_NAKs,0 | Total number of transfers that ended with a NAK (No Acknowledgements) on the previous surfacing. |
| $_XMS_TOUTs,0 | Total number of transfers that ended without a timeout on the previous surfacing. |
| $_SM_DEPTHo,2.36 | Glider-measured depth, in meters, while the glider is at the surface at the end of the previous dive. |
| $_SM_ANGLEo,-58.8 | Glider-measured angle at the surface, at the end of the previous dive, in degrees |
| $GPS2,192327,1910.511, -15645.083,18,1.5,19,9.6 | These values are from the second GPS fix prior to the start of the current dive. See the "Canonical Dive Profile" in the Seaglider Pilot's Guide for further details on where the GPS fix is taken. |
| $SPEED_LIMITS,0.260,0.356 | The minimum and maximum horizontal speed attainable by the Seaglider on this dive, in meters per second. These values are based on the minimum and maximum dive angles and the allowable buoyancyforce. The minimum speed corresponds to the maximum dive angle; the maximum speed is obtained as the minimum value of the horizontal speed. |
| $TGT_NAME,WPT5 | The name of the active target of this dive. See theTargets File section for details. |
| $TGT_LATLONG,2000.000,-15640.000 | The latitude and longitude, in +/-ddmm.mmm and +/- dddmm.mmm format, for the target of this dive. |
| $TGT_RADIUS,1852.000 | The radius for the active target for this dive, in meters. |


$KALMAN_CONTROL,-0.082,0.346
The glider's desired speed to the north, in m/s.
The glider's desired speed to the east, in m/s.
Desired heading is derived from these speeds.
$KALMAN_X,194116.0,-264.0,95.2,-71195.6,1396.7
X displacement from present position to predicted position due to mean, diurnal and semidiurnal components of the model
East position relative to initial position (in meters), at time tk due to glider speed through water
East position relative to initial position (in meters), at time tk due to semidiurnal current
East position relative to initial position (in meters), at time tk due to diurnal current
East position relative to initial position (in meters), at time tk due to mean current
$KALMAN_Y,194116.0,-264.0,95.2,-71195.6,1396.7
Y displacement from present position to predicted position due to mean, diurnal and semidiurnal components of the model
North position relative to initial position (in meters), at time tk due to glider speed through water
North position relative to initial position (in meters), at time tk due to semidiurnal current
North position relative to initial position (in meters), at time tk due to diurnal current
North position relative to initial position (in meters), at time tk due to mean current
$MHEAD_RNG_PITCHd_Wd,337.1,92079,-20.1,-15.000
Desired vertical velocity on dive (cm/s)
Desired vehicle pitch angle
Distance, in meters, to the target
Desired magnetic heading (degrees)

$D_GRID,990 The depth, in meters, to the apogee maneuver, as read from the currently active bathymetry map.

$GCHEAD,st_secs,pitch_ctl,vbd_ctl,depth,ob_vertv, data_pts,end_secs,pitch_secs,roll_secs,vbd_secs, vbd_i,gcphase,pitch_i,roll_i,pitch_ad, roll_ad,vbd_ad,pitch_retries,pitch_errors,roll_retries,roll_errors, vbd_retries,vbd_errors

st_secs: Elapsed time from the start of the dive to the start of the GC

pitch_ctl: Position of the pitch mass, in centimeters, relative to

$C_PITCH (positive aft)

vbd_ctl: Position of the VBD, in cc, relative to $C_VBD (positive buoyant)

depth: Depth at the start of GC, in meters

ob_vertv: Observed vertical velocity, in centimeters per second

data_pts: Number of data records collected thus far in the dive

end_secs: Elapsed time from the start of the dive to the end of the GC

pitch_secs: Number of seconds the pitch motor was on

roll_secs: Number of seconds the roll motor was on

vbd_secs: Number of seconds the VBD was on

vbd_i: Average current used by the VBD, in amps

gcphase: GC phase, encoded as follows

1: Pitch change
2: VBD change
3: Roll
4: Turning (passive)
5: Roll back (to center)
6: Passive mode (waiting)

pitch_i: Average current used by the pitch motor, in amps

roll_i: Average current used by the roll motor, in amps

pitch_ad: Position of the pitch motor, in AD counts, at the end of the motor move

roll_ad: Position of the roll motor, in AD counts, at the end of the motor move

vbd_ad: Position of the VBD, in AD counts, at the end of the motor move

pitch_retries: number of retries (instantaneous AD rate of move less than $PITCH_AD_RATE) during this motor move

pitch_errors: number of pitch motor errors (timeouts) during this motor move

roll_retries: number of retries (instantaneous AD rate of move less than $ROLL_AD_RATE) during this motor move

roll_errors: number of roll motor errors (timeouts) during this motor move

vbd_retries: number of retries (instantaneous AD rate of move less than $VBD_PUMP_AD_RATE_APOGEE, $VBD_PUMP_AD_RATE_SURFACE, or $VBD_BLEED_RATE as appropriate) during this motor move

vbd_errors: number of VBD errors (timeouts) during this motor move

$GC,15,-1.70,-218.4,0.0,0.0,0,59,0.00,0.00,-41.92,0.000,2,0.000,0.000,326,2165,2436,0,0,0,0,0,0

$GC 60,-1.70,-219.0,3.2,-3.9,7,115,11.95,2.45,-34.53,0.000,4,0.180,0.062,2341,3533,3851,0,0,0,0,0,0

$GC,275,-1.70,-219.0,47.8,-22.3,47,281,0.00,2.28,0.00,0.000,6,0.000,0.025,2341,2181,3853,0,0,0,0,0,0

$GC,596,-1.70,-219.0,120.2,-20.3,108,601,0.00,2.50,0.00,0.000,4,0.000,0.045,2341,759,3854,0,0,0,0,0,0

$GC,665,-1.70,-219.0,134.5,-21.4,114,672,0.00,2.33,0.00,0.000,6,0.000,0.023,2342,2149,3855,0,0,0,0,0,0

...lines omitted...

$GC,13111,2.12,489.2,70.3,12.2,557,13165,0.00,2.53,46.45,0.633,4,0.000,0.048,3183,832,959,0,0,0,0,0,0

$GC,13278,2.24,526.9,48.1,13.3,588,13317,0.08,2.38,31.85,0.607,6,0.047,0.025,3213,2229,806,0,0,0,0,0,0

$STATE,20661,end climb,SURFACE_DEPTH_REACHED

$STATE,20661,begin surface coast

$FINISH,1.9,1.008786

- 1.9 — Depth of glider, in meters at the first sample taken after reaching $D_SURF (or $D_FINISH, if enabled)
- 1.008786 — Density of water, in grams per cc, at the first sample taken after reaching $D_SURF (or $D_FINISH, if enabled)

$SM_CCo,2031,75.53,0.653, 0,0,239,530.09

- 2031 — Time in seconds from the start of the dive to when the Surface Maneuver (SM) pump was started
- 75.53 — Time in seconds for the SM pump
- 0.653 — Average current for the VBD during the SM pump, in amps
- 0 — Number of retries during the SM pump
- 0 — Number of errors during the SM pump
- 239 — Final position of the VBD after the SM pump, in AD counts
- 530.09 — Final position of the VBD after the SM pump in cc's

$SM_GC 1.25,11.30,0.00,0.00,0.038,0.000,0.000,424,2272,1263,-10.22,0.34,438.35

$IRIDIUM_FIX,1904.66,12231.77,091207,191902

| | |
|---|---|
| $TT8_MAMPS,0.02301 | Power draw on the 10 V power pack, in amps, measured at the end of the dive. This measurement can be used to determine if devices are being left on. |
| $HUMID,1789 | Pressure inside the pressure hull, in PSIA. |
| $INTERNAL_PRESSURE,7.15848 | Pressure inside the pressure hull, in PSIA. |
| $TCM_TEMP,23.60 | Last temperature reading taken from the compass, in degrees C. |
| $XPDR_PINGS,8 | Number of times the transponder commanded a ping on the dive. This could be altimeter pings, or pings in response to something that sounded like an interrogation. |
| $ALTIM_BOTTOM_PING, 875.1,26.8 | Depth of the glider, and altimeteter-detected distance to bottom. |

$24V_AH,23.3,21.710

Total amp-hours consumed on the 24V battery since the last reset of the battery meters (usually when new batteries are installed).

The minimum measured battery voltage (measured during active phase) on the 24V battery pack, in volts.

$10V_AH,10.0,17.969 Same as $24V_AH, but for 10V battery pack

$FG_AHR_24Vo,6.819

Cumulative A-hr consumed from the 24V battery pack as tracked by the supervisor fuel gauge and recorded at the end of the dive. Only meaningful on a RevC and later motherboard

$FG_AHR_10Vo,6.967 Same as $FG_AHR_24Vo, but for 10V battery pack. Only meaningful on a RevC or later motherboard"

$DEVICES,Pitch_motor,Roll_motor,VBD_pump_during_apogee, VBD_pump_during_surface,VBD_valve,Iridium_during_init, Iridium_during_connect,Iridium_during_xfer, Transponder_ping, Mmodem_TX,Mmodem_RX,GPS,TT8,LPSleep, TT8_Active,TT8_Sampling,TT8_CF8,TT8_Kalman,Analog_circuits, GPS_charging,Compass,RAFOS,Transponder.

Provides the titles of the numbers in the next two lines ($DEVICE_SECS and $DEVICE_MAMPS). The meaning of each title

Pitch_motor: All use of the pitch motor, in the units given in the next two lines
Roll_motor: All use of the roll motor, in the units given in the next two lines
VBD_pump_during_apogee: Use of the VBD pump during active mode
VBD_pump_during_surface: Use of the VBD pump outside of the dive
VBD_valve: Any use of the VBD valve
Iridium_during_init: Use of the phone related to turning the phone on
Iridium_during _connect: Use of the phone while connecting to the basestation
Iridium_during_xfer: Use of phone during a file transfer
Transponder_ping: Use of the transponder during an active ping
Mmodem_TX:
Mmodem_RX:
GPS: All use of the GPS for fix acquisition
TT8: Use of the TT8 at 2 MHz
LPSleep: Use of the TT8 under low power sleep
TT8_Active: Use of the TT8 in active mode
TT8_Sampling: Use of the TT8 while sampling sensors
TT8_CF8: Use of the TT8 while accessing the flash
TT8_Kalman: Use of the TT8 while running the Kalman filter code
Analog_circuits: Use of the analog circuitry, including the pressure sensor
GPS_charging: Use of the auxiliary GPS charging circuit
Compass: Use of the compass
RAFOS: Use of the RAFOS receiver
Transponder: Total use of the transponder (including ping time)

$DEVICE_SECS,28.900,130.775,625.775,0.000,0.000,
32.521,48.298,129.845,2.000,81.068,563.712,9134.856,
711.991,3431.997,344.516,33.374,1911.731,0.000, 3107.613,0.000,0.186

Reports the number of seconds each device was powered on during the dive.

$DEVICE_MAMPS,180.245,87.438,1307.735,0.000,0.000,103.000,160.000,2
23.000,420.000,50.000,19.800,2.190,19.800,
39.800,45.800,81.800,12.000,0.000,8.000,0.000,30.000

Reports the maximum current (in mA) drawn by each device listed in $DEVICES.

$SENSORS,SBE_CT,SBE_O2,WL_BB2F,nil,nil,nil

SBE_CT: Seabird CT sensor. By convention, this is configured as the first device.
SBE_O2: Seabird O2 sensor.
WL_BB2F: Wetlabs BB2F combination backscatter sensor and fluorometer.
Optode: Optode oxygen sensor.
nil: indicates that no sensor is installed in this position.

Similar to $DEVICES, in simply providing titles for the numbers listed in the following two columns ($SENSOR_SECS and $SENSOR_MAMPS). Each title represents one of the sensors installed on the Seaglider, as described here.

$SENSOR_SECS,2182.877,1551.421,748.579,0.000,0.000,0.000

Reports the number of seconds each sensor was powered on during the dive.

$SENSOR_MAMPS,24.000,19.000,105.000,0.000,0.000,0.000

Reports the maximum current drawn by each sensor during the dive.

$DATA_FILE_SIZE,36111,664

- 664: The number of data samples taken during the dive
- 36111: The total size of the data file in bytes

$CFSIZE,260165632,248328192

- 248328192: The available free space on the compact flash card
- 260165632: The total capacity of the compact flash card

$ERRORS, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0

- 0: Number of times the GPS did not provide data from $GPRMC (Position and time) timeout.
- 0: Number of VBD retries
- 0: Number of roll retries
- 0: Number of pitch retries
- 0: Number of VBD errors
- 0: Number of roll errors
- 0: Number of pitch errors
- 0: Number of CF8 retries while closing files
- 0: Number of CF8 retries while writing files
- 0: Number of CF8 retries while opening files
- 0: Number of CF8 errors while closing files
- 0: Number of CF8 errors while writing files
- 0: Number of CF8 errors while opening files
- 0: The number of spurious interrupts. Spurious interrupts may result from divide by zero or memory dereference problems. Th arise from interrupt contention. Occasional isolated spurious interrupts are normal.
- 0: Buffer Overruns-The number of times the log file output is longer than the internal buffer length. For each of the buffer overruns is truncated to fit in the buffer, resulting in lost logfile output.

$GPS,170706,231510,1911.874,-15644.574,40,1.4,40,9.6

- 9.6: Magnetic variation (degrees, positive E)
- 170706: UTC

These values are from the most recent GPS fix, which corresponds to the end of the current dive.

### *2.1.2 Data File*

**(p1230055.dat)**

The .dat file is an ASCII text file that is generated by the Seaglider and transmitted to the basestation for further processing. The first line is the only actual value; all following lines are differences. It serves as the primary conduit for the science data collected by the Seaglider. Each data file covers one dive of information. The format is designed to minimize transmission size and, while clear text, is not intended for direct use by users.

The numbers in the data file can be interpreted by the column titles listed in the "columns" line. The meaning of each column title is summarized below. The first 10 columns ("rec" through "GC_phase") are always present. The remaining columns depend on the sensors installed on the individual glider.

rec: the record number of the individual sample

elaps_t: time since the start of the dive

depth: depth, in centimeters, at the start of the sample

heading: vehicle heading at the start of the sample, in degrees (magnetic)

times 10

pitch: vehicle pitch angle at the start of the sample, in degrees times 10, positive up

roll: vehicle roll at the start of the sample, in degrees times 10, positive starboard wing down

AD_pitch: Pitch mass position, in A/D counts

AD_roll: roll mass position, in A/D counts

AD_vbd: VBD position, in A/D counts

GC_phase: GC phase, encoded as follows

1: Pitch change
2: VBD change
3: Roll
4: Turning
5: Roll back (to center)
6: Passive mode

TempFreq: Temperature, in cycle counts of 4 MHz, in 255 cycles of signal frequency

CondFreq: Conductivity, in cycle counts of 4 MHz, in 255 cycles of signal frequency

redRef: red reference, in A/D counts

redCount: red backscatter, in A/D counts

blueRef: blue reference, in A/D counts

blueCount: blue backscatter, in A/D counts

FluorCount: Fluorometer, A/D counts

VFtemp: BB2F temperature

O2: optional Aanderaa optode oxygen concentration

Temp: optional Aanderaa optode temperature

Dphase: optional Aanderaa optode dphase

### *2.1.3 ASC File*

**(p1230055.asc)**

The .asc, or ASCII, files are created on the basestation. They are essentially the reconstituted (uncompressed, reassembled, and differentially summed) versions of the data (DAT) files created on the

Seaglider. See the Data File section for descriptions of the column names. The entry NaN indicates that there was no sample returned for that sensor. Either the sensor was not installed, or the sensor was not enabled for that sample/deployment, as controlled by the Science File.

### *2.1.4 Eng File*

**(p1230055.eng)**

The .eng, or engineering, files are created on the basestation. They restate data contained in the .asc and .log files, but with the Seaglider control state and attitude observations converted into engineering units. The column titles are described below. The first 10 columns are always present, while the remaining 10 columns vary, depending on the installed sensors.

elaps_t_0000: Time, in seconds, since 0000UTC of the current day
elaps_t: Time, in seconds, since the start of the dive
condFreq: Conductivity frequency, in Hertz.
tempFreq: Temperature frequency, in Hertz.
depth: Depth, in centimeters, at the start of the sample
head: Vehicle heading, in degrees magnetic
pitchAng: Vehicle pitch at the start of the sample, in degrees; positive nose-up
rollAng: Vehicle roll at the start of the sample, in degrees; positive starboard wing down (rolled to starboard)
pitchCtl: Pitch mass position relative to $C_PITCH, in centimeters; positive nose up
rollCtl: Roll mass position, in degrees relative to $C_ROLL_DIVE or $C_ROLL_CLIMB; positive starboard wing down
vbdCC: VBD value relative to $C_VBD, in cc's; positive buoyant
O2Freq: Oxygen concentration (in Hertz)
redRef: Red reference, in A/D counts
redCount: Red backscatter, in A/D counts
blueRef: Blue reference, in A/D counts
blueCount: Blue backscatter, in A/D counts
FluorCount: Fluorometer, in A/D counts
VFtemp: BB2F temperature, in degrees C
O2: Aanderaa optode oxygen concentration
temp: Aanderaa optode temperature
dphase: Aanderaa optode dphase

### *2.1.5 Profiles File*

(p1230055.pro)

The .pro files contain the scientific data that was acquired during the dive, such as temperature and salinity. The column names are as follows:

elapse_time_s_v: time, in seconds, since the beginning of the dive (before the first sample is taken)
Pressure_v: pressure, in decibars
depth_m_v: depth, in meters
TempC_Cor_v: temperature, in degrees C, corrected for 1st order time lag (response time of sensor)
Cond_Cor_v: conductivity, corrected as above
Salinity_v: salinity, calculated

SigmaT_v: density at the current temperature

dive_pos_lat_dd_v: estimated latitude, in decimal degrees. It should be noted that this position is a rough estimate based on the position at the surface, and the depth-averaged current, not an actual GPS or other reading.

dive_pos_lon_dd_v: estimated longitude (see above).

### *2.1.6 Binned Profiles File*

**(p1230055.bpo)**

This is the same data as in the .pro files, but here it is "binned", or averaged, into depth intervals specified by the user.

### *2.1.7 Capture File*

**(p1230055.cap)**

The capture file contains information about all of the actions the Seaglider took during the dive. It captures the output written to the console while the Seaglider is operating. Capture files are a great source of information on the glider's performance, especially in error analysis and debugging. For more information on the use of capture files, please see the Capture File section in the Seaglider Pilot's Guide.

The format of the capture file is not as hard and fast as other file formats, but it usually conforms to that shown below:

time, service, output level, text

Example Capture File

2966.752,N,Capture file opened

- Descriptive text; often what action was taken, and sometimes the reason for the action
- Output level. There are three letters that can appear in this position: N, C, or D. N indicates normal output level. C stands for critical, and means that only output considered critical to glider function will be printed. For the most part, this consists of dire problems with hardware or software, like motor errors or buffer overruns. D stands for Debug, and is used for extended diagnostics.

  This output can be quite voluminous and it is recommended that this only be set on specific services when it is known that some needed output will be captured. Most services do not have output in this level currently, but work is ongoing to add extended diagnostics under this output level.
- Time, in seconds, since the start of the dive

2967.080,HTT8,N,Writing NVRAM...done.

2995.325,HGPS,N,Acquiring GPS fix ...

2998.197,HGPS,N,VVVVA

270407, 140904, 4806.097168, -12222.047852 1.500000 13/13 seconds

3009.584,HTT8,N,Updating parameter $T_GPS_CHARGE to -13320.147

The capture file gives the following information regarding every pitch, roll and VBD maneuver:

876.356,HROLL,N,Roll commanded from 39.80 deg (3384) to 0.00 deg (1976)...

877.415,HROLL,N,34.5 deg (ad: 3195) Updating parameter $R_PORT_OVSHOOT to 18

880.082,SMOTOR,N,MOTOR_DONE: ticks: 1 max 24v: 0.006A avg 24v: 0.006A

880.215,SMOTOR,N,GC TICKS/TIME: 117/119500

880.304,HROLL,N,done.

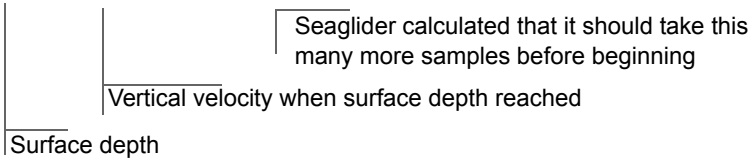
If problems occur, they are reported here.

132.434,HPITCH,N,Pitch completed from -8.87 cm (1472) to -1.24 cm (3130) took 17.0 sec 0.292A (0.377A peak) 97 AD/sec 681 ticks; 10 retries

The capture file also describes changes between dive phases:

839.259,SDIVE,N,Leaving climb state due to SURFACE_DEPTH_REACHED

839.402,SDIVE,N,Entering surface coast state
839.570,SDIVE,N,Reached SD,Wo = 0.064493,6 more points


Seaglider calculated that it should take this many more samples before beginning
Vertical velocity when surface depth reached
Surface depth

### *2.1.8 NetCDF File*

**(p1230055.nc)**

The netCDF file captures all processed files, and is self-documenting. Read-write access to netCDF files is provided by the software libraries supplied by UCAR (University Corporation for Atmospheric Research). The netCDF file is meant primarily for sharing data between scientific users.

### *2.1.9 Private File*

**(p1230055.pvt)**

PVT, or private, files are created on the basestation. They contain data that was originally in the logfile that could pose a security problem if propagated off the basestation (as the logfile may well be). Thus, the data is stripped from the log file and placed in the matched pvt file. The lines in the pvt file correspond with parameters that are listed in the Parameter Reference Manual.

## 2.2 Processing Control Files

This section includes files that are used by the pilot to monitor and, when necessary, modify, how the basestation processes Seaglider data.

### *2.2.1 Communications Log*

**(comm.log)**

The "comm log" file is appended during each communication session, and so is a complete record of the Seaglider's communications over an entire deployment. It is a plain-text file that resides in the Seaglider's home directory. Running tail -f comm.log in the Seaglider's home directory during (or while waiting for) communication sessions is a useful monitor.

Example comm.log

Connected at Sun Dec 2 19:17:03 PST 2007 Date and time of communications session
0055:0:1:0 GPS,031207,031455,1855.179,12237.359,41,1.3,41,-2.1

- Magnetic variation
- Total time to acquire fix. See $N_GPS in the Parameter Reference Manual for details.
- HDOP (Horizontal Dilution Of Precision) - a measure of the strength of the figure used to compute the fix.
- Time to first fix, in seconds
- Longitude, as +/- dddmm.mmm; sign (only minuses are shown; positive East), degrees, minutes, and decimal minutes.
- Latitude, as +/- ddmm.mmm (only minuses are shown; positive North).
- Time, as hhmmss in UTC
- Date, as ddmmyy(after 2000)
- No-comm count: number of calls since last complete data transfer
- Calls made
- Dive number

ver=66.04l,rev=1243M,frag=4,launch=110908,151311
Iridium bars: 5 geolocation: 1846.424805,12238.228516,031207,020210

Location obtained by the iridium phone's geolocation property. This may be accurate to +/- 20km or more.

Sun Dec 2 19:17:20 2007 [sg123] cmdfile/XMODEM: 128 Bytes, 17 BPS
Received cmdfile 17 bytes

Describes the transmission of the command file from the basestation to the Seaglide.

Sun Dec 2 19:17:49 2007 [sg123] sector number = 1, block length = 1024
Sun Dec 2 19:17:54 2007 [sg123] sector number = 2, block length = 1024
Sun Dec 2 19:18:00 2007 [sg123] sector number = 3, block length = 1024
Sun Dec 2 19:18:05 2007 [sg123] sector number = 4, block length = 1024
Sun Dec 2 19:18:07 2007 [sg123] received EOT and read timed out

These lines describe the glider sending a file to the basestation.

- End of transmission

Sun Dec 2 19:18:07 2007 [sg123] sector number = -10,block length = 1024

- Indicates end of file

```
Sun Dec  2 19:18:07 2007 [sg123] done - sending ACK
                                          Acknowledgement that file was sent
Sun Dec  2 19:18:07 2007 [sg123] sg0055lz.x00/XMODEM: 4096 Bytes, 178 BPS
                                   The name of the file is printed after the glider has finished
                                   sending it.
Sun Dec 2 19:18:07 2007 [sg123] Exiting (0)
Sun Dec 2 19:18:14 2007 [sg123] sector number = 1, block length = 1024
Sun Dec 2 19:18:19 2007 [sg123] sector number = 2, block length = 1024
Sun Dec 2 19:18:23 2007 [sg123] sector number = 3, block length = 1024
Sun Dec 2 19:18:28 2007 [sg123] sector number = 4, block length = 1024
Sun Dec 2 19:18:31 2007 [sg123] received EOT and read timed out
Sun Dec 2 19:18:31 2007 [sg123] sector number = -10, block length = 1024
Sun Dec 2 19:18:31 2007 [sg123] done - sending ACK
Sun Dec 2 19:18:31 2007 [sg123] sg0055dz.x00/XMODEM: 4096 Bytes, 189 BPS
Sun Dec 2 19:18:31 2007 [sg123] Exiting (0)
Sun Dec 2 19:18:38 2007 [sg123] sector number = 1, block length = 1024
Sun Dec 2 19:18:43 2007 [sg123] sector number = 2, block length = 1024
Sun Dec 2 19:18:49 2007 [sg123] timeout trying to read next sector
Sun Dec 2 19:18:50 2007 [sg123] finished waiting for next line - cnt = 999
Sun Dec 2 19:18:50 2007 [sg123] got 0x2d sector header
Sun Dec 2 19:18:53 2007 [sg123] finished waiting for next line - cnt  = 746
Sun Dec 2 19:18:54 2007 [sg123] got 0x40 sector header
Sun Dec 2 19:18:57 2007 [sg123] finished waiting for next line - cnt = 787
Sun Dec 2 19:18:59 2007 [sg123] sector number = 3, block length = 128
Sun Dec 2 19:19:02 2007 [sg123] timeout trying to read next sector
Sun Dec 2 19:19:03 2007 [sg123] finished waiting for next line - cnt = 999
Sun Dec 2 19:19:04 2007 [sg123] got 0xe6 sector header
Sun Dec 2 19:19:06 2007 [sg123] finished waiting for next line - cnt = -1
Sun Dec 2 19:19:06 2007 [sg123] got 0xb7 sector header
Sun Dec 2 19:19:07 2007 [sg123] finished waiting for next line - cnt = 875
Sun Dec 2 19:19:08 2007 [sg123] sector number = 4, block length = 128
Sun Dec 2 19:19:10 2007 [sg123] sector number = 4, block length = 128
Sun Dec 2 19:19:10 2007 [sg123] received dup sector = 4
Sun Dec 2 19:19:12 2007 [sg123] timeout trying to read next sector
Sun Dec 2 19:19:13 2007 [sg123] finished waiting for next line - cnt = 999
Sun Dec 2 19:19:13 2007 [sg123] got 0xaf sector header
Sun Dec 2 19:19:15 2007 [sg123] finished waiting for next line - cnt = -1
Sun Dec 2 19:19:15 2007 [sg123] got 0x59 sector header
Sun Dec 2 19:19:17 2007 [sg123] finished waiting for next line - cnt = 543
Sun Dec 2 19:19:17 2007 [sg123] got 0x59 sector header
Sun Dec 2 19:19:21 2007 [sg123] finished waiting for next line - cnt = 130
Sun Dec 2 19:19:23 2007 [sg123] sector number = 6, block length = 128
Sun Dec 2 19:19:23 2007 [sg123] sync error in protocol
```

Errors in transmission are reported. If the Iridium connection drops, the communications session times out.

Duplicate and/or missing sector numbers indicate loss of synchronization between the Seaglider and the basestation. Errors can also be caused by dropped Iridium connections. The Seaglider will automatically call back and try sending data again until it succeeds or reaches the maximum number of calls (set by the parameter $CALL_TRIES).

```
Sun Dec 2 19:19:23 2007 [sg123] sg0055dz.x01/XMODEM: got error
Renamed partial file sg0055dz.x01 to sg0055dz.x01.PARTIAL.1
Sun Dec 2 19:19:23 2007 [sg123] processed partial file sg0055dz.x01 (0x0)
Sun Dec 2 19:19:23 2007 [sg123] Exiting (128)
Disconnected at Sun Dec 2 19:19:39 PST 2007
```

In this case, the glider "realizes" that the basestation did not receive a complete file. The glider will automatically resend the file on the next call.

```
Connected at Sun Dec 2 19:21:39 PST 2007
159:0:2:0 GPS,031207,031455,1855.179,12237.359,41,1.3,41,-2.1
ver=66.03,rev=1243M,frag=4
Iridium bars: 5 geolocation: 1846.424805,12241.375977,031207,070746
Sun Dec 2 19:21:58 2007 [sg123] cmdfile/XMODEM: 128 Bytes, 14 BPS
Received cmdfile 17 bytes
Sun Dec 2 19:22:28 2007 [sg123] sector number = 1, block length = 1024
Sun Dec 2 19:22:33 2007 [sg123] sector number = 2, block length = 1024
Sun Dec 2 19:22:37 2007 [sg123] sector number = 3, block length = 1024
Sun Dec 2 19:22:42 2007 [sg123] sector number = 4, block length = 1024
Sun Dec 2 19:22:45 2007 [sg123] received EOT and read timed out
Sun Dec 2 19:22:45 2007 [sg123] sector number = -10, block length = 1024
Sun Dec 2 19:22:45 2007 [sg123] done - sending ACK
Sun Dec 2 19:22:45 2007 [sg123] sg0055dz.x01/XMODEM: 4096 Bytes, 186 BPS
Sun Dec 2 19:22:45 2007 [sg123] Exiting (0)
Sun Dec 2 19:22:53 2007 [sg123] sector number = 1, block length = 1024
Sun Dec 2 19:22:58 2007 [sg123] sector number = 2, block length = 1024
Sun Dec 2 19:23:03 2007 [sg123] sector number = 3, block length = 1024
Sun Dec 2 19:23:07 2007 [sg123] sector number = 4, block length = 1024
Sun Dec 2 19:23:10 2007 [sg123] received EOT and read timed out
Sun Dec 2 19:23:10 2007 [sg123] sector number = -10, block length = 1024
Sun Dec 2 19:23:10 2007 [sg123] done - sending ACK
Sun Dec 2 19:23:10 2007 [sg123] sg0055dz.x02/XMODEM: 4096 Bytes, 186 BPS
```

If no error is reported, but the basestation does not receive a complete file, the pilot can command the glider to resend the dive by using a Pdos command (see resend_dive in the Extended PicoDos Reference Manual).

The file was successfully resent.

### *2.2.2 SG Calib Constants*

**(sg_calib_constants.m)**

The "calib constants" file contains calibration information about each of the sensors on the Seaglider. This file is created by the pilot or operator, and exists only on the basestation. It does not have a counterpart on the Seaglider. Except for the compass, all of the Seaglider's sensors come calibrated to the Seaglider Fabrication Center.

Their calibration numbers can be found in the notebook delivered with the glider, and should be entered in this file. The compass values are recorded when the Seaglider is fully assembled, and the compass is calibrated in the presence of the batteries and other hardware. The values in this file should be checked, and changed if necessary, whenever new sensors are installed, batteries are changed, or other hardware alterations are made.

The calib_constants file is also used by various visualization tools (Matlab, GLMPC, etc.) to plot Seaglider data. Incorrect values in this file will result in incorrect scientific data in the plots.

Example Calibration Constants File

```
% sg_calib_constants.m
% establishes glider calibration constants

id_str = '128'; Seaglider serial number

mission_title =' Port Susan Aug 15 2007'; pilot or operator specified
   calibcomm =  'SBEs/n0041,  calibration 25 April 2006';
                                  Sensor serial number (found in SG notebook)
                    Sea-Bird Electronics
t_g = 4.37369092e-03 ;
t_h = 6.48722213e-04 ;          SBE temperature sensor calibration coefficients
t_i = 2.63414771e-05 ;
t_j = 2.83524759e-06 ;
% Minimum and maximum frequencies (kHz) for reasonable
% oceanographic values of temperature from SBE calibration
% for C/T s/n 041
sbe_temp_freq_min = 3.214274; % kHz     From SBE sensor calibration. Basestation processing will reject
sbe_temp_freq_max = 6.081845; % kHz     observed temperature frequencies outside of this range.
c_g = -9.97922732e+00 ;
c_h = 1.12270684e+00 ;          SBE conductivity sensor calibration coefficients
c_i = -2.35632554e-03 ;
c_j = 2.37469252e-04 ;
% Minimum and maximum frequencies (kHz) for reasonable
% oceanographic values of conductivity SBE calibration
% for C/T s/n 041

sbe_cond_freq_min = 2.98792; % kHz     From SBE sensor calibration. Basestation processing will
sbe_cond_freq_max = 7.95840; % kHz     reject observed temperature frequencies outside of this range.
cpcor = -9.57e-08 ;
ctcor = 3.25e-06 ;

calibcomm_oxygen = '0106';

Soc = 2.1921e-04;
Boc = 0.0;
Foffset = -825.6362;
TCor = 0.0017;
PCor = 1.350e-04;

mass = 52.173; measured mass of glider

hd_a = 0.003836; lift
hd_b = 0.010078; drag                   Seaglider hydrodynamic parameters
hd_c = 9.8541e-6; induced drag (by lift)
```

```
rho0 = 1027.5;                  Greatest expected water density in area of operation
pitch_min_cnts = 426;
pitch_max_cnts = 3705;
roll_min_cnts = 157;
roll_max_cnts = 3897;           Software limits
vbd_min_cnts = 550;
vbd_max_cnts = 3875;
vbd_cnts_per_cc = -4.0767;
volmax = 51344;                 Volume, in cc, the glider displaces when fully pumped; see the Seaglider Pilot's
                                Guide for tank and ballasting information.
```

### *2.2.3 Pagers File*

**(.pagers)**

The "dot pagers" file controls the automatic notification system. It allows any of three types of messages to be sent to any valid email address: gps, alerts, and recov (see below). This service is run by the data conversion script, which is invoked by a glider logout or disconnection. Lines begininning with a # are comment lines, and are ignored in processing.

```
# Joe Smith
#joe@gmail.com,gps,alerts,recov
jsmith@apl.washington.edu,recov
2065551234@messaging.sprintpcs.com,recov
# Jane Jones
jjones@apl.washington.edu,gps,alerts,recov
```

Joe Smith will receive emails to his APL account, and text messages to his Sprint phone, but will not receive messages to his gmail account.

If the glider goes into recovery, send the most recent GPS position and the recov code.

Send an alert when the basestation has a problem converting a file or files.

After every dive, send the most recent GPS position and, if the glider is in recovery, the recov code.

```
#2063335555@vtext.com,gps,alerts,recov
#2061239999@vtext.com,gps,alerts
#Iridium Phone
#881645559999@msg.iridium.com,gps
```

### *2.2.4 .URLS*

**(.urls)**

The "Dot URLs" file is read by the basestation, following processing of dive data (triggered by a Seaglider logout. It specifies URLs on which to run GET for each processed dive. This can be used for any supported httpd function, and is mainly used to poll for data transfers to support visualization servers. The first entry on

the line is the timeout (in seconds) to wait for a response to the GET. It is separated from the URL by a tab. convert.pl adds arguments "instrument_name=sg&dive=" with the proper separator. Comments in the file are indicated by a #

Example .urls file

1 http://iop.apl.washington.edu/~glider/cgi-bin/update.cgi

### *2.2.5 Basestation Log*

baselog_000000999999,baselog.log)

Time; hhmmss (time zone as kept on basestation)

Date; ddmmyy

The baselog_ file is produced by the basestation, and logs the output from the scripts that perform the data conversion and notification functions of the basestation. It is written during each invocation. This file is the first place to look when debugging problems with the data conversion. If the basestation cannot process a file, it sends an alert to any contact listed in the .pagers file.

The baselog.log is an accumulation of all of the basestation conversions reported in the baselog_ files, without the timestamps.

## 2.3 On-board Glider Information

This section includes files that are stored on the Seaglider. Most of the information in these files is used by the glider in calculations regarding navigation and energy usage.

### *2.2.6 Processed Files Cache*

**(processed_files.cache)**

This file contains the dives that have been processed and the time of processing. To force a file to be re-processed, delete the corresponding line from this file. Comment lines are indicated by a #.

Example processed_files.cache

# Written 14:54:28 23 Feb 2008 UTC

st0007pz.000, 19:05:58 21 Feb 2008 UTC

sg0000kl, 14:54:28 23 Feb 2008 UTC

st0007du, 19:05:58 21 Feb 2008 UTC

st0007lu, 19:05:58 21 Feb 2008 UTC

st0009du, 19:40:22 21 Feb 2008 UTC

st0009kz, 19:16:44 21 Feb 2008 UTC

st0009lu, 19:37:51 21 Feb 2008 UTC

st0010du, 20:21:33 21 Feb 2008 UTC

st0010kz, 20:15:35 21 Feb 2008 UTC

st0010lu, 20:15:34 21 Feb 2008 UTC

st0011du, 14:54:28 23 Feb 2008 UTC

st0011kz, 14:30:35 23 Feb 2008 UTC

st0011lu, 14:30:35 23 Feb 2008 UTC

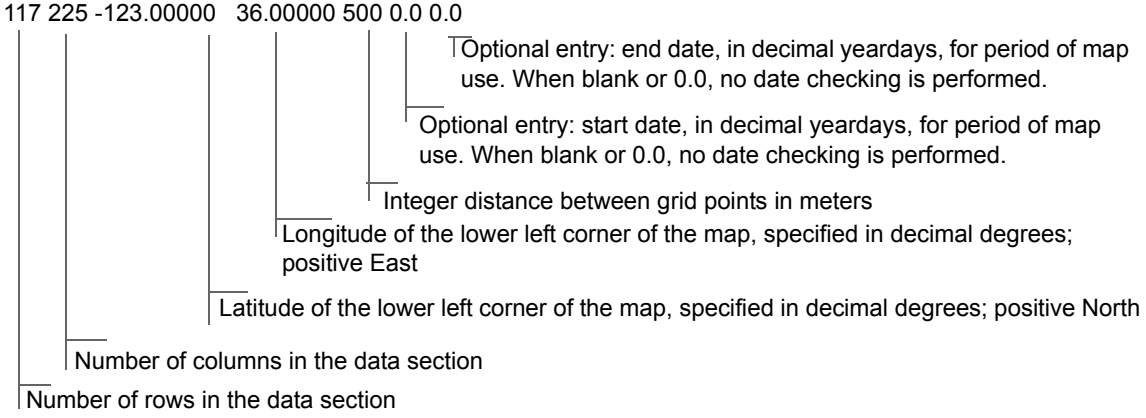
### *2.3.1 Bathymap*

When the bathymetry map-reading function of the glider is enabled, this file contains the map. It is usually uploaded to the Seaglider's compact flash before deployment, but may be uploaded in the field if necessary.

Map files provide the glider with geographic (and sometimes temporal) environmental information. A bathymetry map provides the glider with bathymetry data about a given region of the ocean. The glider may carry up to 999 bathymetry maps (the files are named bathymap.000), but in practice far fewer are on board. These maps are not required for gliders to fly. For more details on how bathymetry maps are used, see the Navigation section of the Seaglider Pilot's Guide.

In addition to bathymetry maps, the glider can carry ice maps which indicate a spatially and temporally varying climatology of ice cover. The glider can use this information to make decisions about surfacing.

Both kinds of maps contain a fixed-size header, followed by a variable-length data section. The header is defined as follows:


117 225 -123.00000 36.00000 500 0.0 0.0
Optional entry: end date, in decimal yeardays, for period of map use. When blank or 0.0, no date checking is performed.
Optional entry: start date, in decimal yeardays, for period of map use. When blank or 0.0, no date checking is performed.
Integer distance between grid points in meters
Longitude of the lower left corner of the map, specified in decimal degrees; positive East
Latitude of the lower left corner of the map, specified in decimal degrees; positive North
Number of columns in the data section
Number of rows in the data section

For a bathymetry map, the data section contains the depth of the bottom at each grid point, expressed in integer meters. The data is stored in column major order.

For an ice map the data section contains ice condition values for the time period between the start and end dates at each grid point. Ice condition values are stored as 2-bit integers packed sequentially together into sixteen equal length periods spanning the dates between start date and end date. Valid condition codes are: 0 = always surface, 1 = possibly ice, 2 = probably ice, 3 = always ice. As an example, for a start date = 0.0

and end date = 365.0, the lowest two bits of the value at any grid point encode the ice condition for the first 23 days of the year. Bits 2 and 3 cover the condition for the next 23 days, etc.

### *2.3.2 Battery File*

**(BATTERY)**

The Battery File is used by the glider to keep track of power consumption throughout the time the glider is using the battery pack. The Battery File is not intended to be edited by the user.

Pitch_motor 3041.069

Amp seconds drawn by this device since the battery pack power tracking was initiated

See $DEVICES and $SENSORS in the Log File section of this document.

Roll_motor 990.029
VBD_pump_during_apogee 216074.641
VBD_pump_during_surface 82015.531
VBD_valve 0.000
Iridium_during_init 17540.021
Iridium_during_connect 9597.448
Iridium_during_xfer 48699.711
Transponder_ping 873.774
Mmodem_TX 0.000
Mmodem_RX 0.000
GPS 5227.668
TT8 11375.065
LPSleep 3565.161
TT8_Active 9204.906
TT8_Sampling 30932.490
TT8_CF8 25142.061
TT8_Kalman 2861.964
Analog_circuits 10045.106
GPS_charging 0.000
Compass 5552.722
RAFOS 0.000
Transponder 126.060
SBE_CT 5738.196
SBE_O2 4966.481
WL_BB2F 59876.422

### *2.3.3 Compass Calibration File*

(TCM2MAT).123

Seaglider serial number

The compass is calibrated in the assembled glider, to account for effects of the metal on the compass readings. This file is stored on the glider by the assembler, and is not intended to be edited by the user.

Example Compass Calibration File

tcm2mat.sparton_SN100.sg123.080807

Date of last calibration (ddmmyy)

Seaglider serial number

compass type and serial number

-0.0184 0.8424 0.1660 0.0466
0.0133 0.9603 0.0447 -0.0185
0.0984 -0.0018 0.0018 0.0010 0.1054 -0.0004 -0.0008 0.0012 0.1040
53.9472 -17.3493 5.8241 compass calibration values

### *2.3.4 Capvec File*

The Capvec File is parsed by the glider and updates one or more elements of the Capture Vector. Normally, this file is not used except for glider provisioning. See the capvec and parse_capvecfile commands in Extended PicoDOS Reference Manual for details on updating the Capture Vector, and the section Capture Files in the Seaglider Pilot's Guide for details how and when to use capture files.

The Capvec File is a line oriented format. Lines may be comment lines, in which case the first character must be a /. All other lines are updates to the Capture Vector and are documented under the capvec command in the Extended PicoDOS Reference Manual.

## 2.4 Command and Control Files

These files are created by the pilot to control the Seaglider mission characteristics. Formats are given here, but usage of these files is discussed in the Seaglider Pilot's Guide.

### *2.4.1 Targets File*

**(targets)**

The Pilot creates the targets file. One target is listed per line, and the target name must be listed first. The order of the other fields does not matter. Comments can be included, preceded by a %.

| | | | | |
|---|---|---|---|---|
| SEVEN | lat=4807.0 | lon=-12223.0 | radius=200 | goto=SIX |
| SIX | lat=4806.0 | lon=-12222.0 | radius=200 | goto=FIVE |
| FIVE | lat=4805.0 | lon=-12221.0 | radius=200 | goto=EIGHT |
| FOUR | lat=4804.0 | lon=-12220.0 | radius=200 | goto=EIGHT |

| KAYAKPT | lat=4808.0 | lon=-12223.0 | radius=100 | goto=KAYAKPT |
|---|---|---|---|---|
| Target name - this can be any string of numbers and/or letters, without whitespace. | Latitude, in +/- ddmm.m; positive North | Longitude, in +/- dddmm.m; positive East | Radius, in meters, within which the Seaglider determines it has reached the target. | Next target - this target name must be specified in the Target column. |

Above is a typical version 66 targets file. It has all the fields necessary to direct the Seaglider to targets. There are also four optional fields, which can be added as columns in the targets file:

| escape=KAYAKPT | depth=100 | finish=90 | timeout=3.0 |
|---|---|---|---|
| The escape_target specifies what target to move to if the glider has been unable to navigate for a specified length of time (e.g. if it is stuck under the ice). The escape_target must be a valid named target in the file and can vary for each named target. One possible future use is to have the standard targets along a cyclical survey route all point to a single escape target that then points (through next_target) to a series of targets that define an entire route to a convenient recovery location. | Specifying a value for depth on a target means that target can be achieved by crossing a bathymetric contour. If the value is positive the target is achieved when crossing that contour from deep to shallow. When negative, target achievement is defined by moving across that contour from shallow to deep. The glider measures its depth for comparison against the target depth either by altimetry or via a $T_NO_W timeout during the dive phase. | Finish specifies a direction (degrees), and establishes a finish line through the target, perpendicular to the direction specified. The target is considered achieved when the difference between the bearing to the target and the finish direction is greater than 90 (or less than -90) degrees. Example 1: finish direction of 90 specifies a north-south finish line drawn through the target; the target is achieved when the glider is east of the line. Example 2: finish direction of 180 specifies an east-west finish line; target is achieved when glider is south of the line. A value of -1 or no specification of finish means that no finish line will be tested. | Timeout specifies a length of time (in days) that the glider should try to achieve this target. If the timeout is exceeded the glider will proceed to the target named by goto. If timeout is not specified or is given as zero then the glider will try to achieve the target with no time limit. |

### *2.4.2 Science File*

**(science)**

This file, created by the pilot, contains instructions for the Seaglider about when to sample with the scientific instruments. Comment lines are indicated by a/, and columns are separated by tabs.

Example Science File

// Science for Port Susan

The bottom limit of each depth bin

The most frequent sample interval in this depth bin

Each digit in this column corresponds to one sensor. Sensors and sensor order vary by Seaglider. Consult $SENSORS in the Log File. Multiply this digit by the number in the time column to calculate how often this sensor should sample in this depth bin.

The time interval which controls how often the guidance and control algorithms are run.

| /depth | time | sample | gcint | |
|---|---|---|---|---|
| 20 | 6 | 100 | 60 | ——This row indicates that from the surface (0 meters) to 20 meters, the first sensor should sample every 6 seconds. The second and third sensors should be turned off. During Guidance and Control, all sensors should sample every 60 seconds. |
| 50 | 12 | 100 | 180 | |
| 200 | 12 | 120 | 300 | —— This row indicates that from 50 to 200 meters, the first sensor should Sample every 12 seconds, the second should sample every 24 seconds, and the third sensor should be turned off. During Guidance and Control, all sensors should sample every 300 seconds. |

### *2.4.3 Command File*

**(cmdfile)**

Refer to the Pilot's Guide for more information on the Command File.

### *2.4.4 Pdos Commands File*

**(pdoscmds.bat)**

The file pdoscmds.bat is created by the pilot, and uploaded to the Seaglider. It is used to deal with the Seaglider's software. See the Extended PicoDOS Reference Manual for information.

APPENDIX C

# *Extended PicoDOS® Reference Manual, v66.07*

This appendix contains the *Extended PicoDOS Reference Manual*, v66.07, revised December 2009.

You can use the copy command in addition to the commands in this reference:

**copy <source filename> <destination filename>**

Description: This command copies a source file to a destination specified.

# *Extended PicoDOS® Reference Manual*

*SCHOOL OF OCEANOGRAPHY and APPLIED PHYSICS*
*LABORATORY UNIVERSITY OF WASHINGTON v66.07*
*December 2009*

# Chapter 1
# Introduction and Conventions

## 1.1 Introduction

This manual is a reference to extensions to the PicoDOS® operating system command set implemented in the Seaglider operating code. PicoDOS® is a registered trademark of Persistor Instruments, Inc., Bourne MA, USA.

The version number of this document coincides with the version of the main Seaglider operating code in which these extensions exist.

These extensions either make new functions available from the PicoDOS® prompt, or extend the capabilities of existing functions. Commands are only available through the Seaglider operating code, which intercepts and interprets the commands, passing them to PicoDOS® as appropriate. While in this mode, the Seaglider code passes any command not explicitly recognized as an extended PicoDOS® command on to PicoDOS® itself for execution. Limited error reporting exists in this case.

The extensions are accessible at the PicoDOS® prompt available from the main menu when connected directly to a Seaglider (exceptions as noted below), or by uploading the **pdoscmds.bat** file when the Seaglider is operating autonomously. In the former case, the Seaglider code displays a standard PicoDOS® prompt (picoDOS>) with an extra ' >', as follows.

picoDOS>>

In the latter case, results of the extended PicoDOS® commands are captured to a file and transferred to the Seaglider basestation (in

compressed form, named **sg0055pz.000**, using "0055" as a placeholder for dive number, and "000" for increment number) following execution of the commands. The basestation renames this file per the **p1230055.000.pdos** convention.

The **pdoscmds.bat** file is a plain text sequence of extended PicoDOS® commands, one per line. Lines that begin with a forward slash ('/') are interpreted as comments and are ignored.

Chapter 2 is a list of the extended PicoDOS® commands, grouped by functional area. Chapter 3 is an alphabetical-order reference for each command. Note that standard PicoDOS® commands are not documented here.

## 1.2 Document Conventions

Extended PicoDOS® commands are shown in ***bold italic*** type below. Options and arguments are shown on the same line. Exposition follows in plain type. File names are given in **lower-case bold** font. Click on the name of a command to jump to its description. Use the "Back" button to return to the previous location.

## 1.3 PicoDOS®

PicoDos® is Persistor Instrument's DOS-like operating system for the CF8/TT8 combination used on Seaglider. It provides access to the DOS FAT file system on the Compact Flash, as well as some simple file manipulation utilities. The TOM8 and PicoDOS® commands are documented in the PicoDOS® User's Guide of November, 2000 (which is incorporated into this document by reference). The extensions below provide additional functionality, either to extend PicoDOS® generally or to provide Seaglider-specific functions.

# Chapter 2.

# List of Extended PicoDOS® Commands

## Help and exit

?
??
pdos
tom8
quit

## File manipulation and data

***bathycat***
***<filespec>+ [[> | >>] <outfile>]***
***[del | rm] [/v] <filespec>+***
***gunzip <zipped_file> <file> gzip <file> <zipped_file>***
***md5 [<signature>] <file>***
***[ren | rename] <file1> <file2>***
***resend_dive [/l| /d | /c | /t] <dive> <fragment>***
***science***
***split <filename>***
***strip1a <filename> [<size>]***
***sumasc <file>***
***tar [c | x] <file> <filespec>***
***[xs | put | xr | get] <filespec>+***

## Control

***$PARAM,value***
***clearnv <passphrase>***
***dumpnv***
***readnv <varname>***
***reboot <run_file> <arguments>***
***[target | targets] [<new_target> [<radius>]]***
***writenv <varname> <value>***

## Diagnostics

***capvec [<service> <level> <dest>]***
***flash_errors***
***menu <menuspec> [<arguments>]***

***parse_capvecfile <file>***
***usage***
***ver***

# Chapter 3

# Extended PicoDOS® Command Reference

### *bathy*

Causes the on-board bathymetry files to be re-read, checked, and, if appropriate, loaded into memory. Useful for checking the integrity of the bathymetry files on the compact flash.

### *capvec [<service> <level> <dest>]*

Without arguments, displays the current capture vector. The capture vector describes the capture output level and destination for each of the Seaglider's hardware and software services. Every output line in the Seaglider source code that is capturable is assigned a service and an output level (verbosity). The capture vector controls where the output is routed to and what the output level (verbosity) is for each service. With arguments, sets the capture vector for a specific service (table 2.1) to the specified level (table 2.2) and destination (table 2.3).

**TABLE 2.2. Available services for capturing**

| Service | Description |
|---|---|
| HPITCH | Pitch motor |
| HROLL | Roll motor |
| HVBD | VBD Pump and Valve |
| HPHONE | Modem hardware |
| HGPS | GPS receiver |
| HTT8 | TT8 Computer |
| HCF8 | Flash hardware |
| HANALOG | Analog circuits and control |
| HCOMPASS | Compass hardware |
| HRAFOS | RAFOS hardware |
| HSBECT | Seabird CT sensor |
| HSBEO2 | Seabird O2 sensor |

**TABLE 2.2. Available services for capturing (Continued)**

| Service | Description |
|---|---|
| HWLBB2F | Wetlabs sensor |
| HOPTODE | Optode O2 sensor |
| HBATT | Battery hardware and charging |
| HPRES | Pressure Sensor |
| HXPDR | Transponder hardware |
| SPOWER | Software managing power |
| SBATHY | Software managing bathymetry |
| SNAV | Software managing navigation (primarily targets) |
| SKALMAN | Software kalman filter |
| SMOTOR | Software controlling all motor movements (primarily GC nterrupt handler) |
| SSENSOR | Software controlling all sensors (primarily data sampling) |
| SDIVE | Software controlling dive and flight |
| SSURF | Software controlling surface activities |
| SEPDOS | Extended PicoDOS support |
| SSYS | Software utilities and infrastructure |
| SUSR | Software dealing primarily with human console interaction (primarily tests and menus) |
| SGLMALLOC | Seaglider's heap implementation |

**TABLE 2.3. Levels at which output can be captured for a given service**

| Output Level | Description |
|---|---|
| CRITICAL | Only trigger output thaty is critical in nature - usually associated with an extreme hardware problem (such as motor errors) or a critical software problem - such as buffer or heap overrun. |

**TABLE 2.3. Levels at which output can be captured for a given service**

| Output Level | Description |
|---|---|
| NORMAL | The vast majority of Seaglider output falls into this category currently. |
| DEBIG | Extended diagnostics. This output can be quite volumnious and it is recommended that this only be set on specific services when it is known that some needed output will be captured. Most services do not have output in this level currently, but work is ongoing to add extended dignostics under this output level. |

**TABLE 2.4. Available destinations for capture output of any service**

| Destination | Description |
|---|---|
| **NONE** | Do no output for this service. |
| **PRINT** | Send output only to the operator console. |
| **FILE** | Send output only to the **capture file**. |
| **BOTH** | Send output to both the operator console and to the **capture file**. |

Note that a higher level of output also implies the lower levels. For example, setting a services output to **NORMAL** implies that **CRITICAL** output will also be triggered.

Example: *capvec HVBD DEBUG PRINT*

Sets the hardware VBD service output to debug level (most verbose) and routes the output for that service to the operators console only.

By default, the Seaglider software sets the output levels to **NORMAL** and output destination to **BOTH** for all services. As a side effect of this change, the **capture file** is almost always being filled with output.

***cat <filespec>+ [ [> | > >]<outfile>]***
Concatenates files in the same general way as the standard Unix cat command. Admits use of wildcards (*) in the file specification.

Example: *cat chunk.U30 chunk.U31 > > chunk.GZ*

***clearnv <passphrase>***
Clears contents of non-volatile RAM utility storage. The passphrase must exactly match one of the strings hard-coded in epdos.c: I_really_mean_it! clears all utility storage (including password and telephone numbers), I_mean_it! only clears latched target and flight state information.

***[del | rm] [/v] <filespec>+***
Extends the standard delete (del) command by admitting use of wildcards (*) in the file specification.

Example: *del SG01*LZ.A*

***dumpnv***
Dumps contents of non-volatile RAM.

***flash_errors***
Reports CF8 file open, write and close retries and errors.

***gunzip <zipped_file> <file>***
Uncompresses file compressed with gzip.

Example: *gunzip chunk.GZ MAIN.RUN*

***gzip <file> <zipped_file>***
Compresses file with gzip.

Example: *gzip MAIN.RUN MAIN.GZ*

***lsleep <seconds>***
Pause execution in low power sleep for up to 60 seconds.

***md5 [<signature>] <file>***
Generates 128-bit md5 hashes of the specified file. If a signature (hash) is specified, md5 compares the specified signature (hash) with the one it generates for the file and generates an error if they are not identical (in the character-by-character sense). Used to verify the integrity of files uploaded to the Seaglider. In particular, md5 is the verification part of the protocol used to upload, verify and

reboot new executable code on the Seaglider.

Example: *md5 082ab2b60d626181e73b17429c55dd8f chunk.GZ*

### *menu <menuspec> [<arguments>]*

Execute commands from the Seaglider code menu tree, by specifying the absolute menu path to the command and any required arguments. The menu is specified by menu names, separated by forward slashes, '/'. The arguments are specified in a whitespace-separated list: arg1=val1 arg2=val2 ... .

Example: *menu /hw/ct*

### *$PARAM,value*

Changes the specified parameter to the specified value. Parameters are specified by three-digit numbers (*nnn*), and are in lineal order as they appear in the Seaglider menus (or in the code source file *parms.h*).

Example: *$T_DIVE,330*

### *parse_capvecfile <file>*

Parses a capture vector file and updates the capture vector. This command is normally used during glider provisioning or testing - the preferred way to set the capture vector is through the capvec command.

Example: *parse_capvecfile capvec.new*

### *pdos*

Exits the Seaglider code to native picoDOS® on the TT8/CF8.

### *quit*

Exits the Seaglider extended picoDOS® mode and returns to the Seaglider main menu.

### *readnv <varname>*

Read the value of the specified variable, where the variable is one of the following: *target_name, password, selftest_count, boot_count, last_known_lon, last_known_lat, last_last_fix_time, magvar, fly_escape_route, fly_safe_depth, device0, device1, device2, device3, telnum*, or *altnum*.

### *reboot <run_file>*

Reboots the Seaglider using the specified file name as the executable file. Note that the .run suffix is not used in this command. It is important to verify that the file that is the target executable is not corrupted. A way to prevent unrecoverable hangs is to never rename new unproven executables **main.run**. Name them

**mainnew.run**, for example, and issue the ***reboot*** command as ***reboot mainnew***. If the Seaglider hangs during the reboot, the watchdog timer should initiate another reboot, but to the executable code called *main*, which presumably was running when the ***reboot*** command was issued. This is a fallback safety feature and should be noted.

Example: *reboot REVA*

### *[ren | rename] <file1> <file2>*

Rename command which calls the CF8 rename command directly, and is aliased to work with either ***ren*** or ***rename***.

Example: *ren REVA.RUN MAIN.RUN*

### *resend_dive [ /l | /d | /c | /t ] <dive> <fragment>*

The mechanism by which the pilot can ask the Seaglider to resend individual pieces of previous log (/l), data (/d), capture (/c) or tar (/t) files. This is used to recover missing pieces of data, which are most often caused by protocol confusion between the Seaglider and the basestation (on whether or not a particular data piece was successfully transferred). The data are specified by type, dive number, and fragment number (the third 4kB piece of dive 123's data file would be specified "/d 123 2" (data piece counter starts at 0). Leading zeros on the fragment number are optional.

Example: *resend_dive /d 289 1*

### *science*

Parses the on-board **science file**. Used mainly in the laboratory or test tank to establish sampling intervals for testing.

### *sleep <seconds>*

Pause execution for up to 60 seconds.

### *split <file>*

Splits files into pieces of size **$N_FILEKB** (kBytes).

Example: *split sg0115dz.r*

### *strip1a <file>[<size>]*

Strips trailing XMODEM padding characters ('1a') off the end of a file, optionally to a specified size (in bytes).

Example: *strip1a chunk.U31 1898*

### *sumasc <file>*

Sums succeeding lines of data file to recreate original observed values. This is the inverse of the simple successive difference compression scheme used for glider data files.

### *tar [c | x] <tarfile> <filespec>+*

Implementation of standard Unix-style tar (tape archive) utility. Admits wild-card (*) expressions.

Example: *tar c data.tar SG01*DZ.A*

### *[target | targets] [<new_target> [<radius>]]*

The mechanism by which the pilot can change the current Seaglider target and the target radius from among the targets listed in the current **targets file** on the Seaglider. The **new_target** must be in the list of targets in the **targets file** on the Seaglider flash. The radius is in meters.

Example: target NE_CORNER
Example: target SW_CORNER 1852

### *tom8*

Exits the Seaglider main program and native picoDOS® to the TT8 monitor program, TOM8. PicoDOS is started by issuing the command 'go 2bcf8' at the TOM8 prompt.

### *usage*

Provides a summary of disk usage on the compact flash.

### *ver*

Provides versioning and configuration information for software and hardware installed (and running) on the Seaglider.

### *[xs | put | xr | get] <file1> <file2>*

XMODEM protocol file transfer commands (issued from the glider).

Example: xr chunk.U30
Example: xs sg0150DZ.A

File names for these commands must be specified in the dos 8.3 file name specification. If a file name does not conform to these specifications and error will be returned that the file name is too long.

### *writenv <varname> <value>*

Write specified value of specified variable to non-volatile RAM (NVRAM) utility storage. Variables stored in NVRAM are *target_name, password, selftest_count, boot_count, last_known_lon, last_known_lat, last_last_fix_time, magvar, fly_escape_route, fly_safe_depth, device0, device1, device2, device3, telnum*, and *altnum*.

### *?*

Top-level help command, which only displays methods of exiting the Seaglider code's PicoDOS® mode, and documents the ***??*** command.

### *??*

Extensive help command, which displays methods of exiting the Seaglider code's PicoDOS® mode, lists the available extended PicoDOS® commands, and gives usage hints.

Copyright University of Washington, 2006

APPENDIX D

# *Dive Data Visualization Software*

## *Introduction*

This Appendix will describe how to execute the software that analyzes dive data acquired from the sensors installed on a Seaglider; also to indicate to the user the basic description of each plot displayed when the visualization scripts are run.

## *Acronyms, Abbreviations, and Definitions*

| **Term** | **Definition** |
|---|---|
| CT | Conductivity, Temperature, Depth sensor |
| GPCTD | Glider Payload Conductivity, Temperature, Depth sensor |
| PAR | Photosynthetically Active Radiation |
| GUI | Graphical User Interface |
| UI | User Interface |
| IDE | Integrated Development Environment |

## *Required Tools*

### Software

- MATLAB 2010a (Other versions may work but are not supported or recommended)
- iRobot Seaglider Dive Data Analysis Software with Seawater Library 1.2d (visualization software)

### Hardware

- Any computer or portable device capable of running MATLAB2010a and displaying plots.

## *Installation and Configuration*

### MATLAB 2010a

Install MATLAB 2010a per the instructions that are provided with the MATLAB 2010a software.

### iRobot Dive Data Analysis Software (Visualization Software)

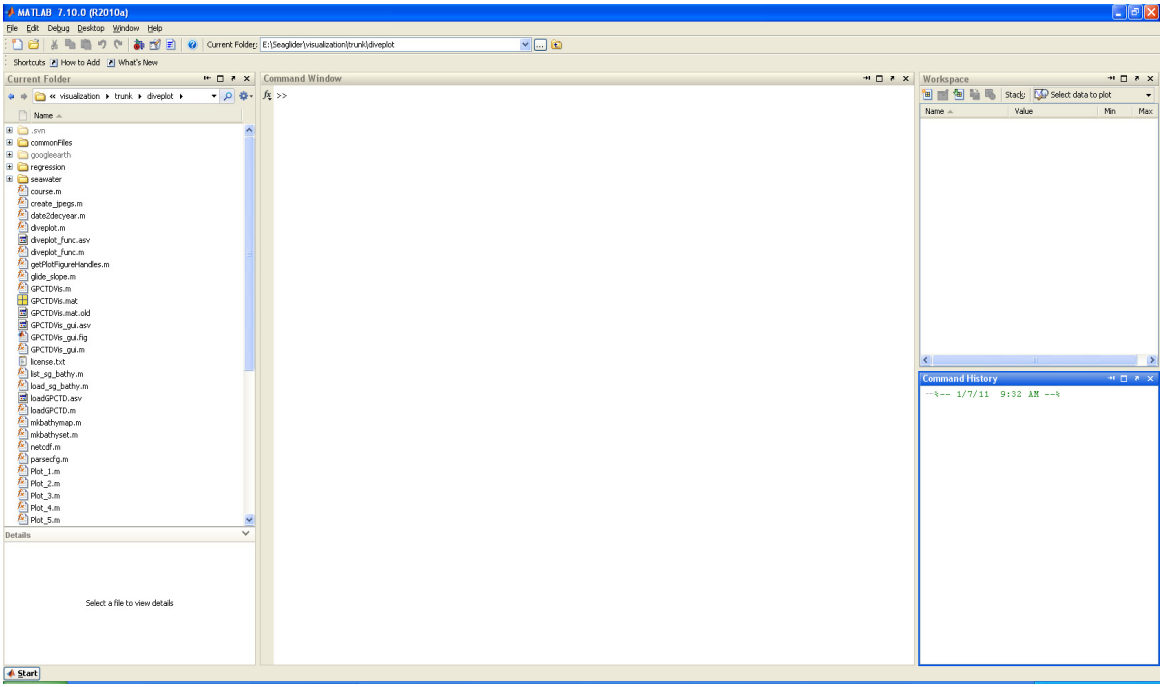
The Dive Data Analysis Software comes packaged in a compressed archive file (*.zip). The contents must be extracted to your computer in order to use the software.

1. Create a folder on your root drive and name it, **DivePlot** (or a more appropriate name pertaining to the specific mission).
2. Copy the visualization files supplied under the “Matlab Scripts” folder in your CD (or via FTP site) into the DivePlot folder.

## *Using the Dive Plot Analysis Software*

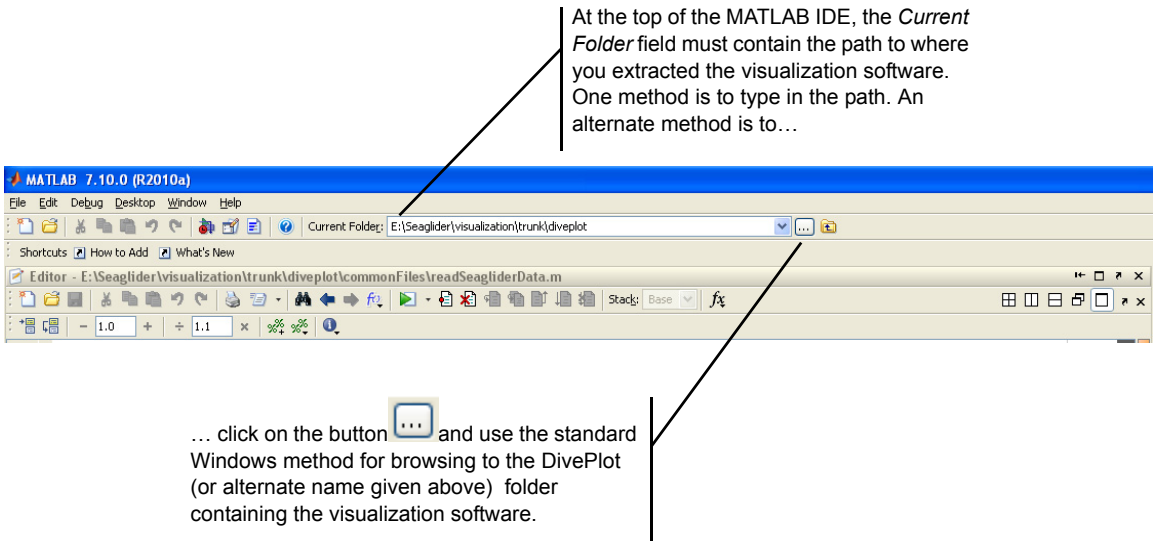
### Run the MATLAB Application

Per the instructions provided by MATLAB 2010a, start the MATLAB software. When the MATLAB Integrated Development Environment (IDE) appears on your computer monitor, it will look very similar to the following screen capture:



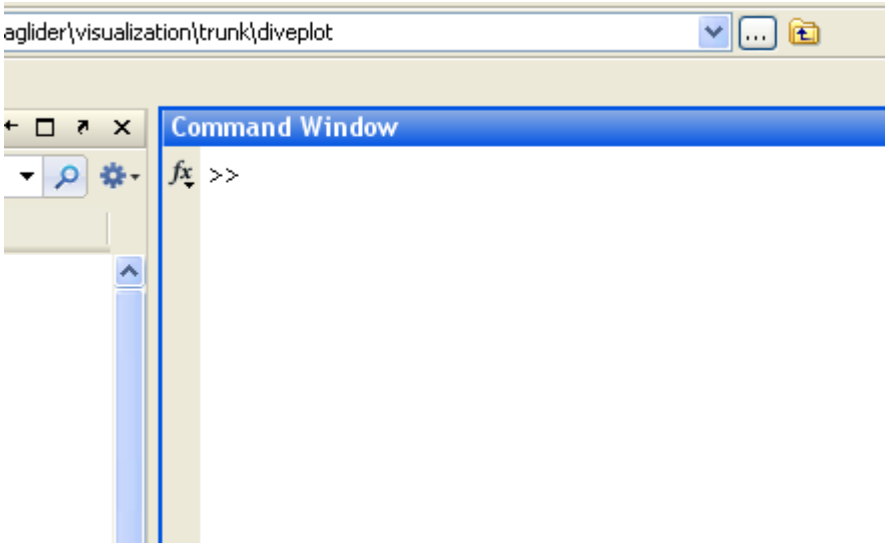

## Set Current Folder

Once the MATLAB IDE is displayed on your computer monitor, the *current folder* must be set to the folder created called **Dive Plot** (or alternate name given above) containing the visualization software. Below is an illustration.

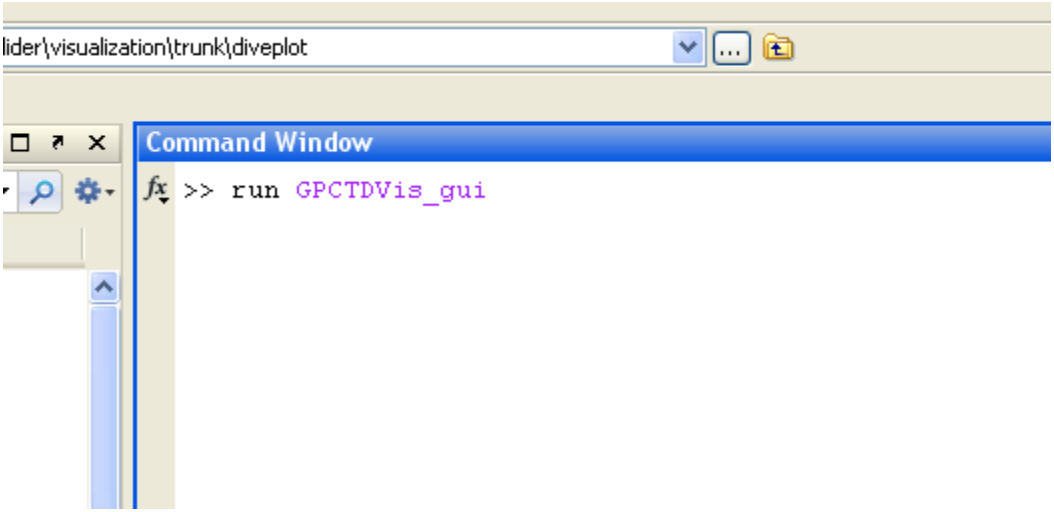

At the top of the MATLAB IDE, the *Current Folder* field must contain the path to where you extracted the visualization software. One method is to type in the path. An alternate method is to…
MATLAB 7.10.0 (R2010a)
File Edit Debug Desktop Window Help
Current Folder: E:\Seaglider\visualization\trunk\diveplot
Shortcuts How to Add What's New
Editor - E:\Seaglider\visualization\trunk\diveplot\commonFiles\readSeagliderData.m
Stack: Base
… click on the button and use the standard Windows method for browsing to the DivePlot (or alternate name given above) folder containing the visualization software.

## Loading the Dive Plot GUI

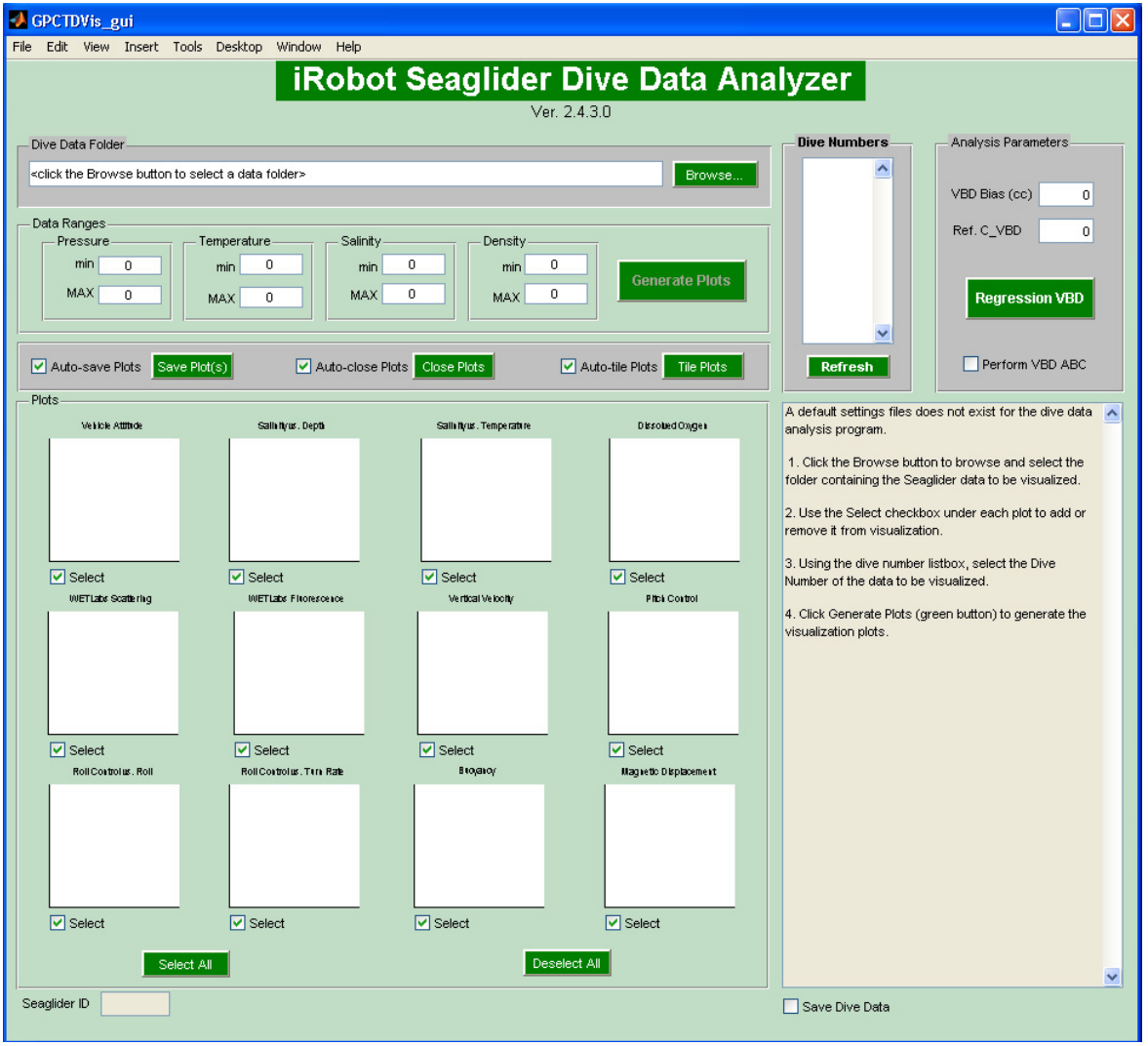
The MATLAB IDE contains a window titled, *Command Window*. Within the *Command Window* is a command prompt, >>. Refer to the following diagram to familiarize yourself with this window which will be used in the steps that follow.



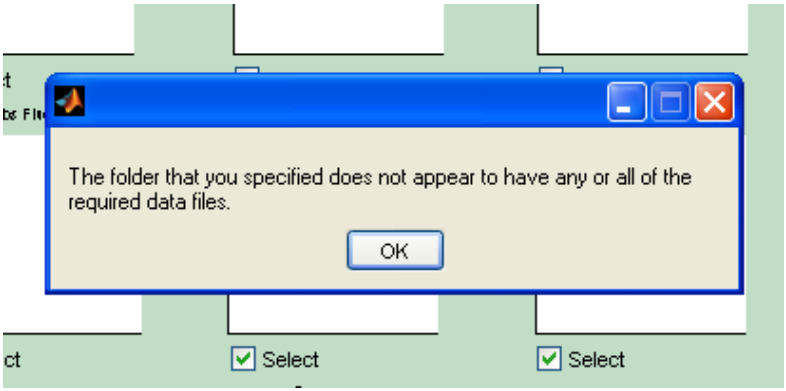
1. Move the mouse cursor to the right of the command prompt and click and release the left mouse button. You should see a blinking cursor – typically a vertical line or bar – next to the command prompt.
2. Type the following command at the command prompt: **run GPCTDVis_gui**




3. Press the **Enter** key on your keyboard. In several seconds you will see the Dive Data Analysis graphical user interface (GUI) appear. If this is the first time running the visualization software on the computer, the GUI will look like the following figure.




Also when running for the first time, a notification stating the folder specified does not have the data required to run software; the notification may be hidden behind the GUI screen. This is because the path that appears in the *Dive Data Folder* field upon startup of GUI is an invalid path and meant to be changed. Simply close this window by selecting ‘OK’ and begin to browse to the appropriate **DivePlot** (or alternate name given above) folder.




## Analyzing Dive Data

### *Specifying the Dive Data Folder*

Analyzing dive data begins with selecting the **DivePlot** folder that contains the mission dive data. Typically you will want to have the data on your local computer so that the analysis is performed quickly.

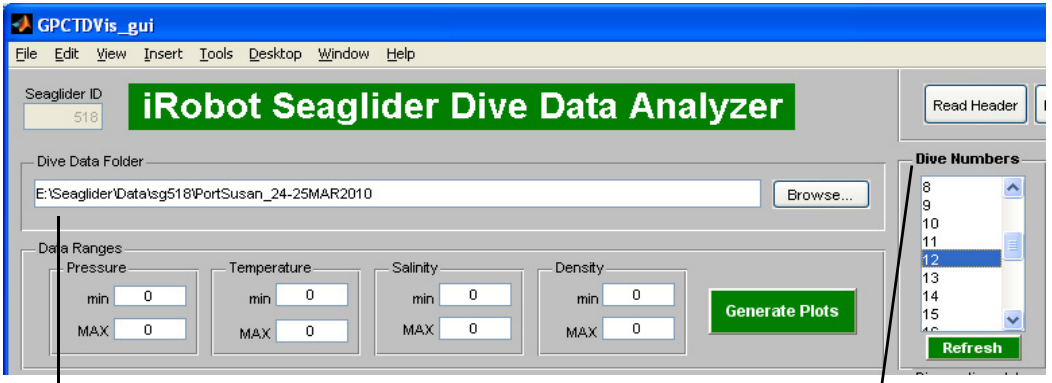
Specifying the data folder in the Dive Data Analysis GUI is similar to selecting the Current Folder in the MATLAB IDE as described in section “Set Current Folder” on page 317. You can either type the pathname into the *Dive Data Folder* field or click on the **Browse** button and use the standard Windows method of browsing to a folder. Refer to the following diagram to assist with performing these tasks.

Either type the full path to the folder containing the dive data, or…

… click the Browse button and use the standard Windows dialog box to navigate to the folder containing the dive data.

### *Selecting Dive Data for Analysis*

The Dive Data Analysis GUI has a field titled, *Dive Numbers.* Once the folder containing dive data has been selected, the *Dive Numbers* field will be populated with the available dives. This is a typical Windows type field where you use the scrollbar to scroll up or down the list to the dive number to be analyzed, and then click on the dive number in the field to select the dive for analysis.




Once you have specified the folder containing the dive data files…

... the Dive Numbers field will be populated with a list of available dives.

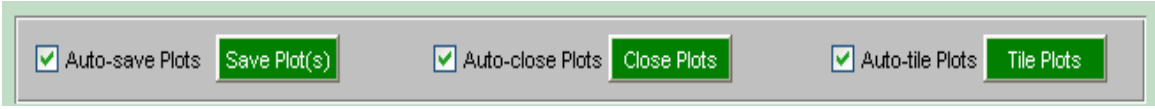
### *Performing the Analysis*

The Dive Data GUI provides the capability to customize the actions of the software when the data analysis is performed. These options will be discussed in more detail in subsequent sections of this document. For now, we will use the default values provided by the software.

Performing the analysis of the dive data for the *Dive Number* selected is accomplished by simply clicking the Generate Plots button on the Dive Data GUI. Upon clicking the button, the software will begin analyzing the data and generating the graphs to visualize the data for analysis.

**Auto-Save, Auto-Close, and Auto-Tile Plots**

The following section of the Seaglider Dive Data GUI allows you to perform several actions once you begin to generate plots.




*Auto-save plots* will create a folder containing all visualization plots, in jpeg format, for the specified dive number.

If the Auto-save Plots box is checked and the user then clicks the Generate Plots button, a folder will be automatically placed within the **DivePlot** folder where all mission dive data has been stored containing all selected plots. T

he folder is titled *pxxxyyyy_DivePlots*; where xxx is the Seaglider serial number and yyyy is the dive number (e.g. p5050002_DivePlots). If the box was not checked when the Generate Plots button is selected, the plots can still be saved after they are generated by selecting Save Plot(s).

The *Auto-close plot* automatically closes any open plots. The

button can be clicked to close plots, or if the *Auto-close Plots* box is checked as in example above, the plots will also close if another *Dive Number* is plotted using the

button.

*Auto-tile Plots* aids in organizing visualization plots on the user's computer screen. If the Auto-tile Plots box is checked, and the user then clicks the

button, the plots will be automatically organized in a window pane style on the user's screen automatically. If the *Auto-tile Plots* box is not checked when the

button is selected, the plots will be stacked one on top of the other, with plot 1 being first, but can still be organized after generation by selecting

. An example screen shot of the *Auto-tile Plot* function is seen below.

## *Visualizations*

Multiple graphs are displayed for the variety of data that is collected and reported by the vehicle. Each type of data has a specific graph that is associated with the data. So, for instance, data collected from one or more oxygen sensors will be plotted on the same graph. The following subsections will discuss in detail each data type analyzed and visualized by the software.

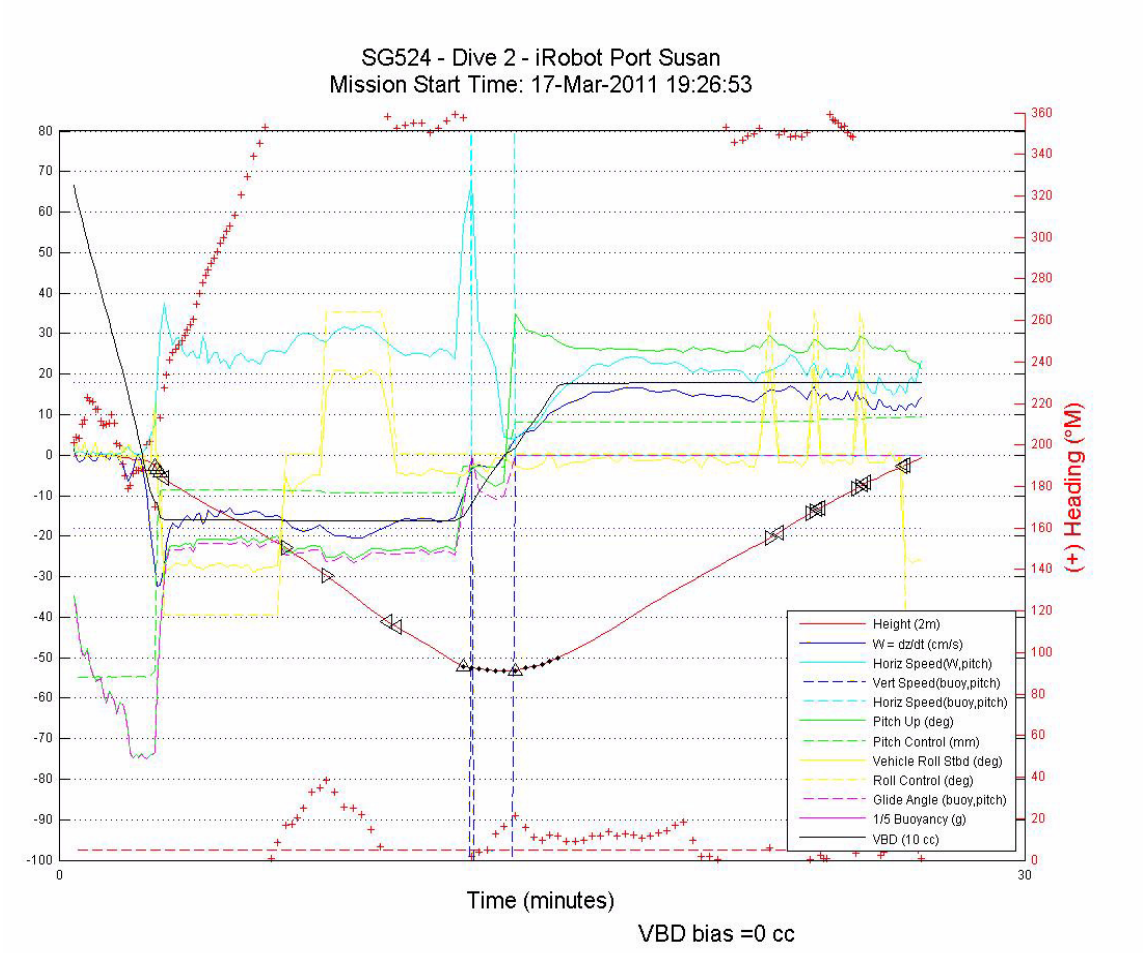
Graphs for which no data exists will display a message on the graph indicating that no data exists for the dive being analyzed. So for example, if no oxygen data exists for the dive a message such as “Oxygen data does not exist for this dive,” will be displayed on the graph. Refer to the subsections below for messages specific to the plot. If the *sg_calib_constants.m* file is missing or the input/format is incorrect, the plots may not plot as intended; In this event, the Dive Data Analysis Software may display the raw data or may not show the data at all. Review *sg_calib_constants.m* file for correctness before generating plots.

**Plot 1: Composite Plot**

Plot 1 presents a composite graph including the following dive characteristics:

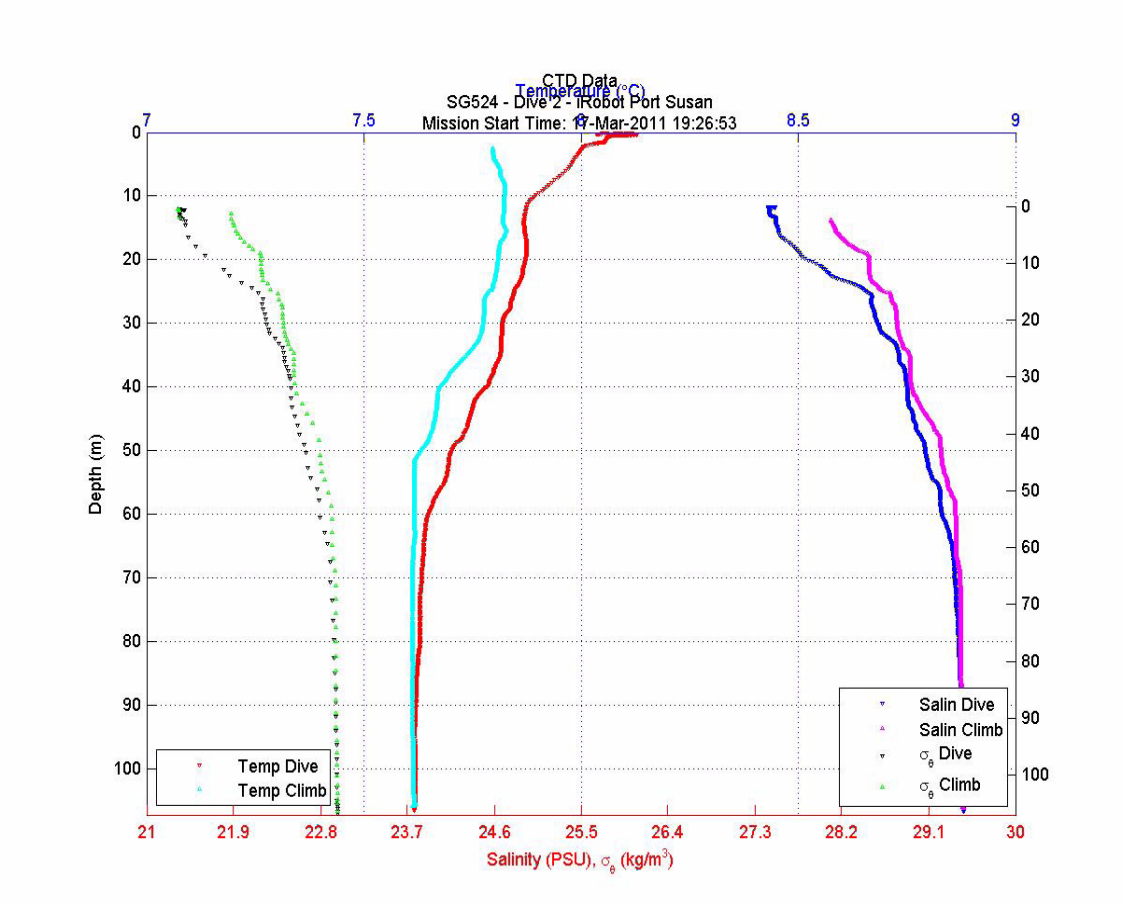
- Velocities (Horizontal and Vertical). Horizontal velocity is represented by the horizontal speed in the direction of heading. Vertical velocity is presented as a magnitude in the z-direction (perpendicular to the horizon).
- Dive Depth
- Buoyancy
- Pitch/Roll
- Heading

Other than acting as a general flight check, this plot primarily aids in providing a value to the pilot for changes in the parameter $C_VBD. This can be determined by looking at the vertical velocity (w = dz/dt) versus the VBD. Rule of Thumb: If VBD line crosses 0 depth before w line, glider is heavy; if glider is too heavy, decrease $C_VBD. And vice versa: If w line crosses 0 depth before VBD line, glider is light; if glider is too light, increase $C_VBD.

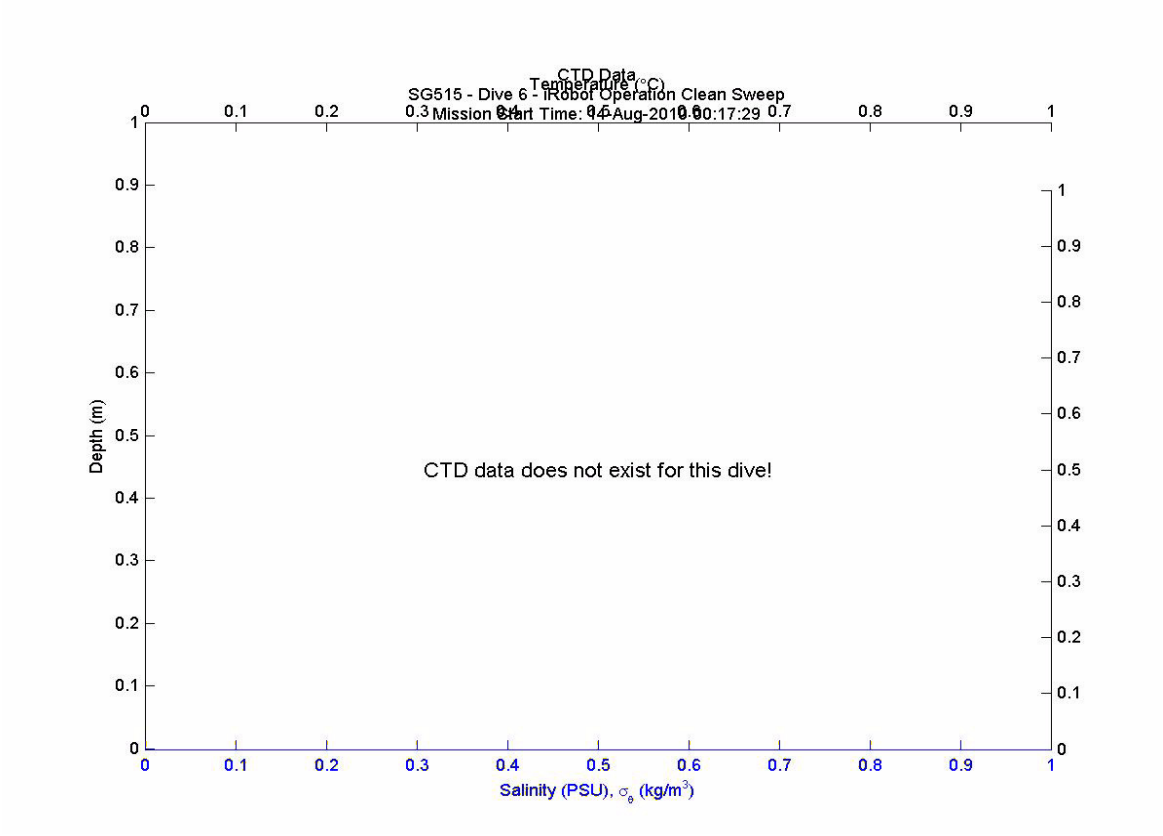

SG524 - Dive 2 - iRobot Port Susan
Mission Start Time: 17-Mar-2011 19:26:53
(+) Heading (°M)
Time (minutes)
VBD bias =0 cc
Height (2m)
W = dz/dt (cm/s)
Horiz Speed(W,pitch)
Vert Speed(buoy,pitch)
Horiz Speed(buoy,pitch)
Pitch Up (deg)
Pitch Control (mm)
Vehicle Roll Stbd (deg)
Roll Control (deg)
Glide Angle (buoy,pitch)
1/5 Buoyancy (g)
VBD (10 cc)

**Plot 2: CTD Plot (from Sea-Bird CT Sail or Sea-Bird GPTCD)**

Plot 2 presents CTD data as Depth vs. Salinity and Temperature on both the descent and ascent. The depth data is measured through the glider's pressure sensor. Several colors are used to denote temperature and salinity on descent and ascent. PSU indicates practical salinity units.

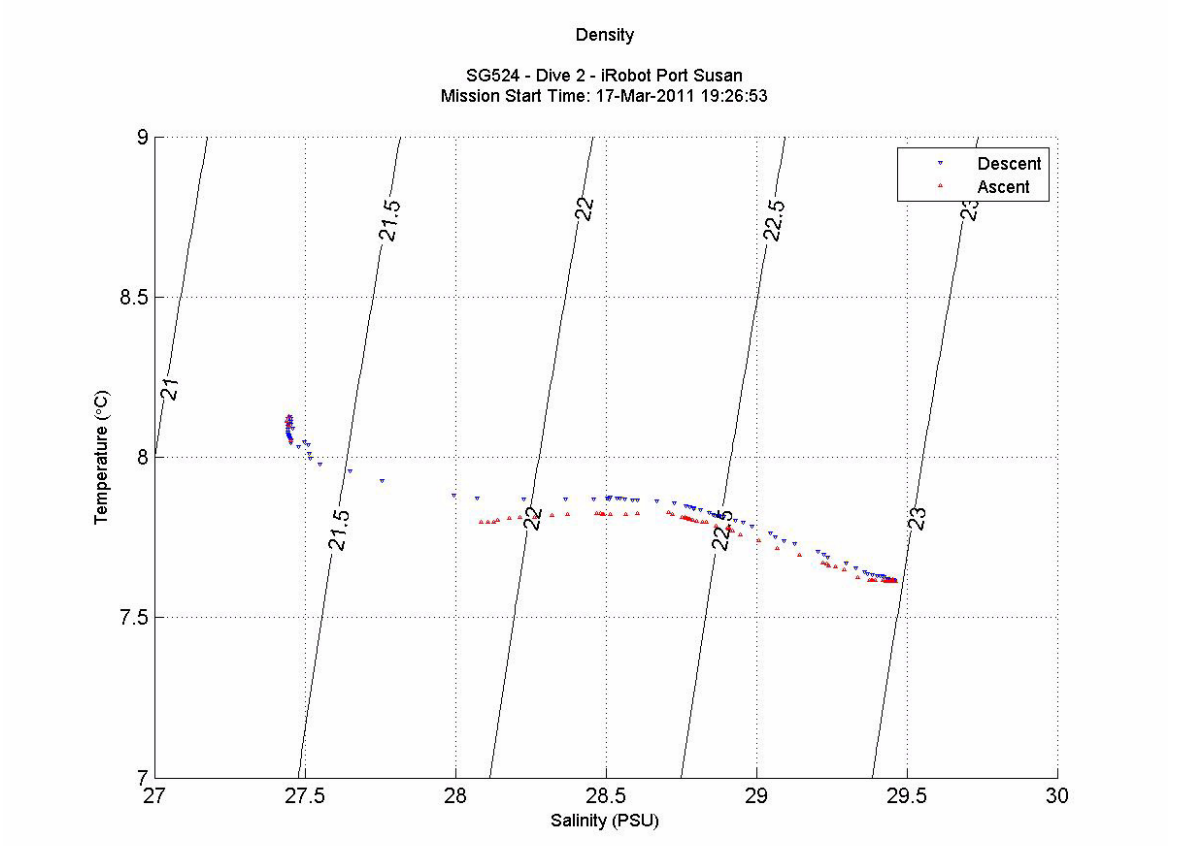



**Note:**
When CTD data is not present or sg_calib_constants.m file is incorrect, the message "CTD data does not exist for this dive!" will appear as seen in example below.


CTD Data
Temperature (°C)
SG515 - Dive 6 - iRobot Operation Clean Sweep
Mission Start Time: 14-Aug-2010 00:17:29
CTD data does not exist for this dive!
Depth (m)
Salinity (PSU), σθ (kg/m3)

**Plot 3: T-S Plot: (from Sea-Bird CT Sail or Sea-Bird GPTCD)**

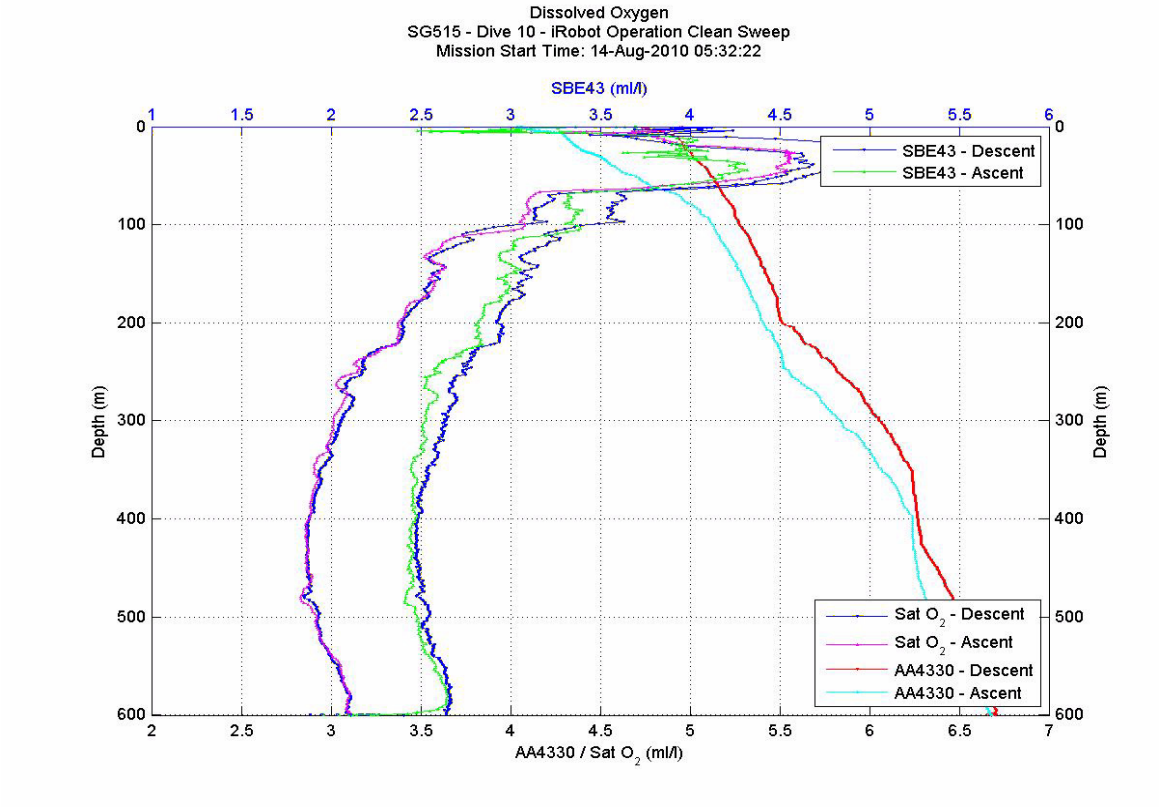
The Temperature-Salinity (T-S) Plot displays the relationship between temperature, salinity and density. Density is displayed on the plot as isopycnal curves. Densities are plotted in different colors for both the dive descent and ascent. PSU indicates practical salinity units.

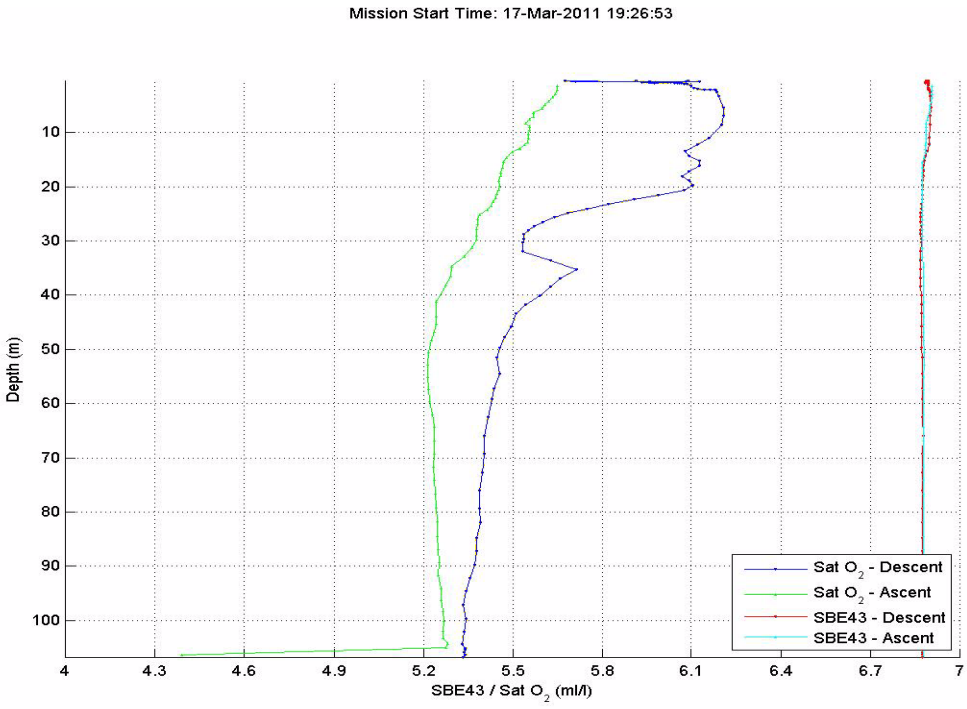


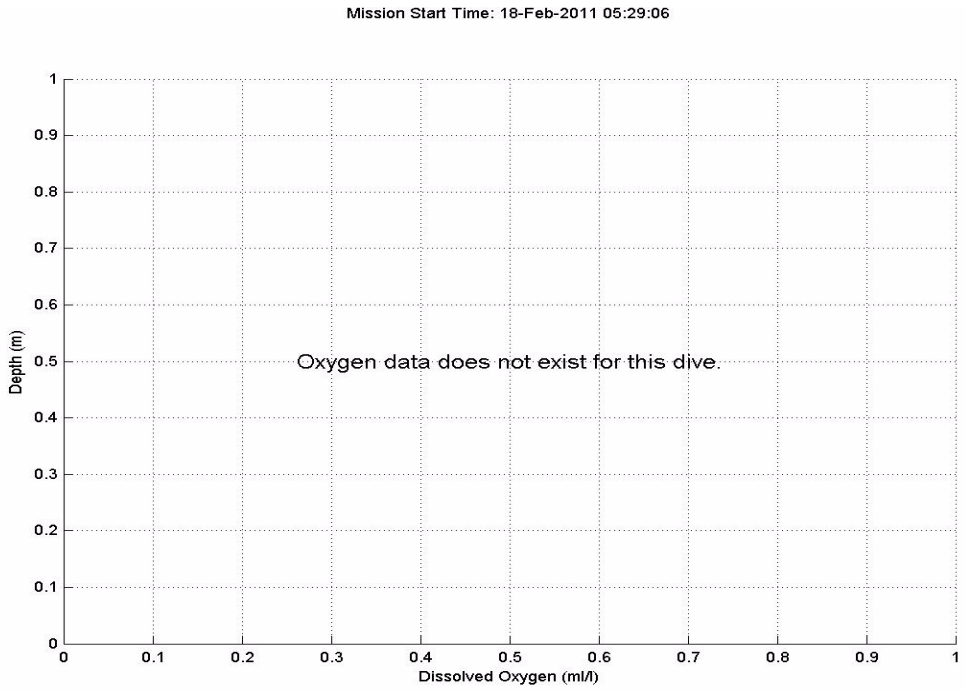
**Plot 4: Dissolved Oxygen Plot (SBE DO, and/or Aanderaa DO Sensor)**

The dissolved oxygen plot will plot all oxygen data collected during the dive. Oxygen data collected will be plotted using colors to distinguish data gathered on the descent and ascent. Examples of plots below; one plot is a configured glider with two dissolved oxygen (DO) sensors, one plot is a configured glider with 1 DO sensor, and 1 is a configured glider with no DO sensor.

If oxygen data is available, the graph will also display oxygen saturation data as well. Oxygen saturation data is calculated using the Seawater library (part of the MATLAB scripts). The Seawater library uses salinity, temperature, and pressure for this calculation and does ***not*** require oxygen data acquired from DO sensor. It is simply added to the DO plot as it is more relevant here than on the CTD plot; therefore, the oxygen saturation data will only be plotted if oxygen data is available to plot. If no oxygen data is available, the plot will read "Oxygen data does not exist for this dive" and the oxygen saturation will not be plotted.

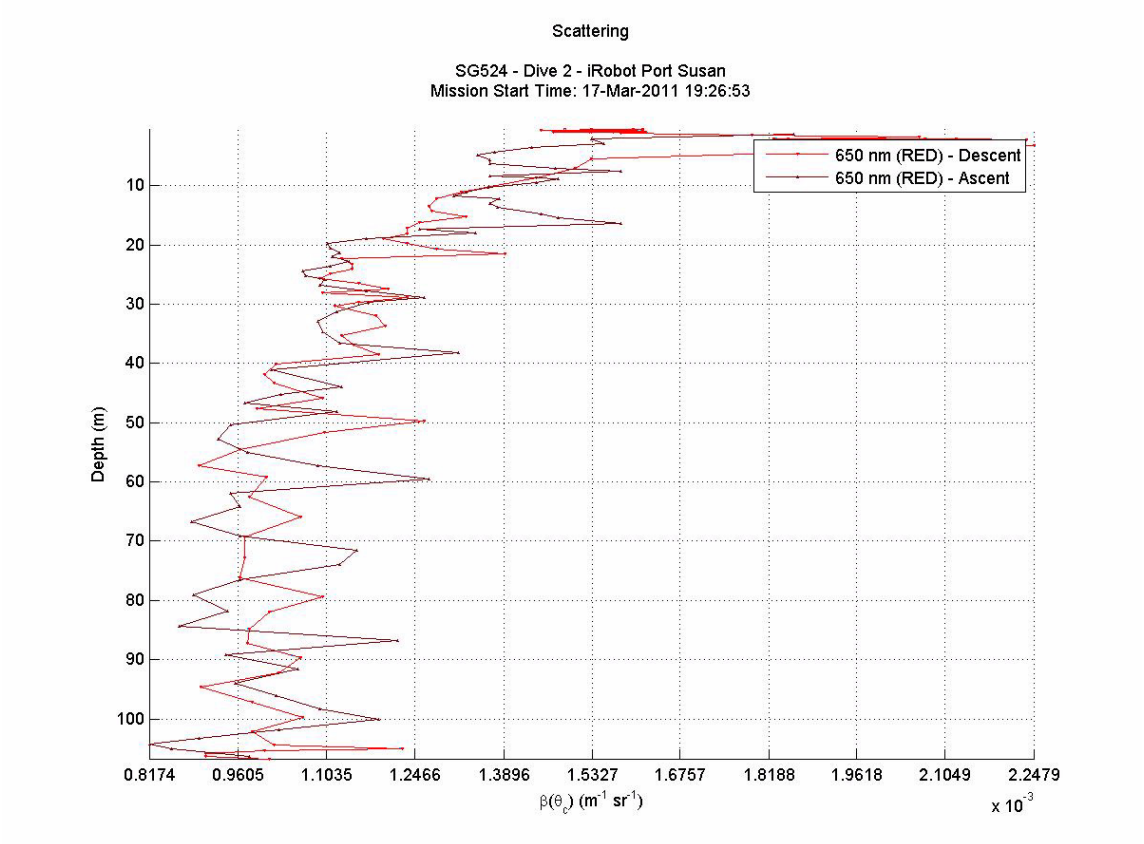

Dissolved Oxygen
SG515 - Dive 10 - iRobot Operation Clean Sweep
Mission Start Time: 14-Aug-2010 05:32:22
SBE43 (ml/l)
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
SBE43 - Descent
SBE43 - Ascent
Depth (m)
0
100
200
300
400
500
600
Sat O2 - Descent
Sat O2 - Ascent
AA4330 - Descent
AA4330 - Ascent
2
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7
AA4330 / Sat O2 (ml/l)

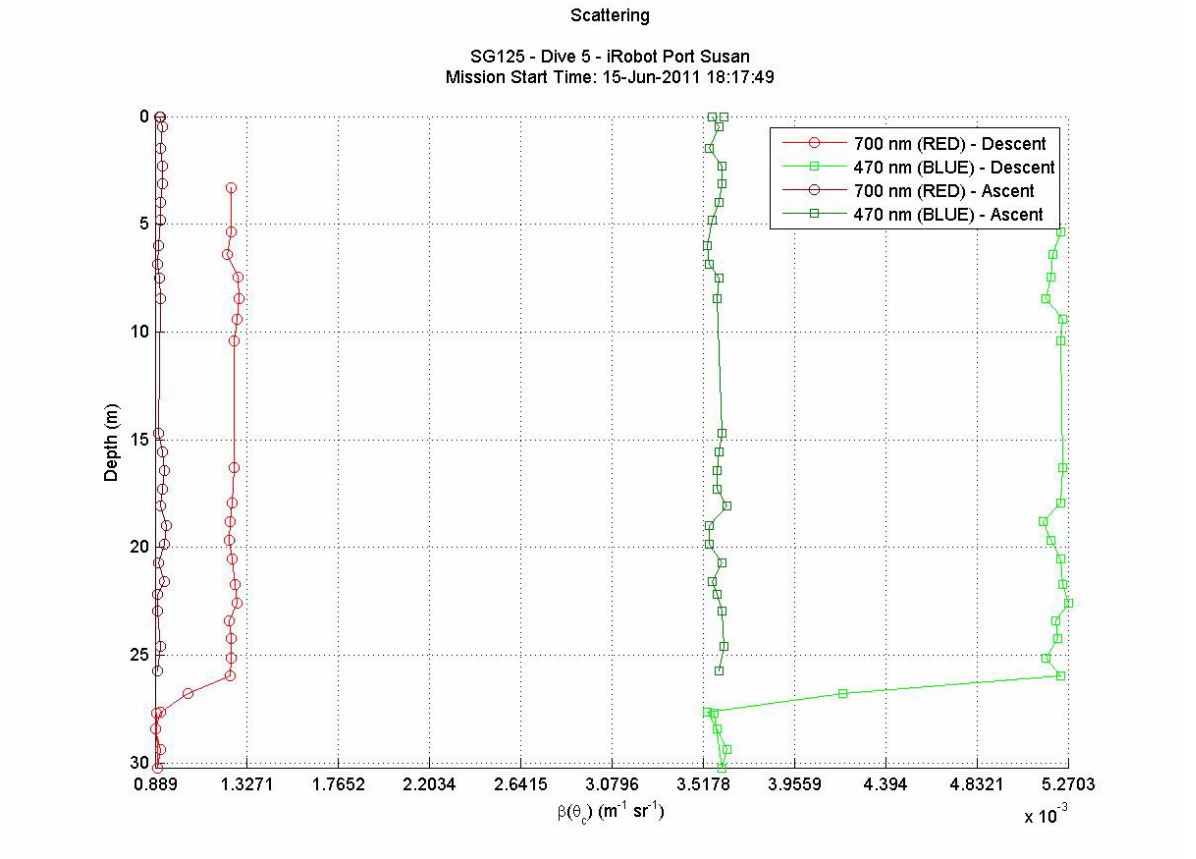

Mission Start Time: 17-Mar-2011 19:26:53
Depth (m)
10
20
30
40
50
60
70
80
90
100
4
4.3
4.6
4.9
5.2
5.5
5.8
6.1
6.4
6.7
7
SBE43 / Sat O2 (ml/l)
Sat O2 - Descent
Sat O2 - Ascent
SBE43 - Descent
SBE43 - Ascent


Mission Start Time: 18-Feb-2011 05:29:06
Oxygen data does not exist for this dive.
Depth (m)
Dissolved Oxygen (ml/l)
0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
0.9
1

### Plot 5: Backscatter Plot (WET Labs Triplets)

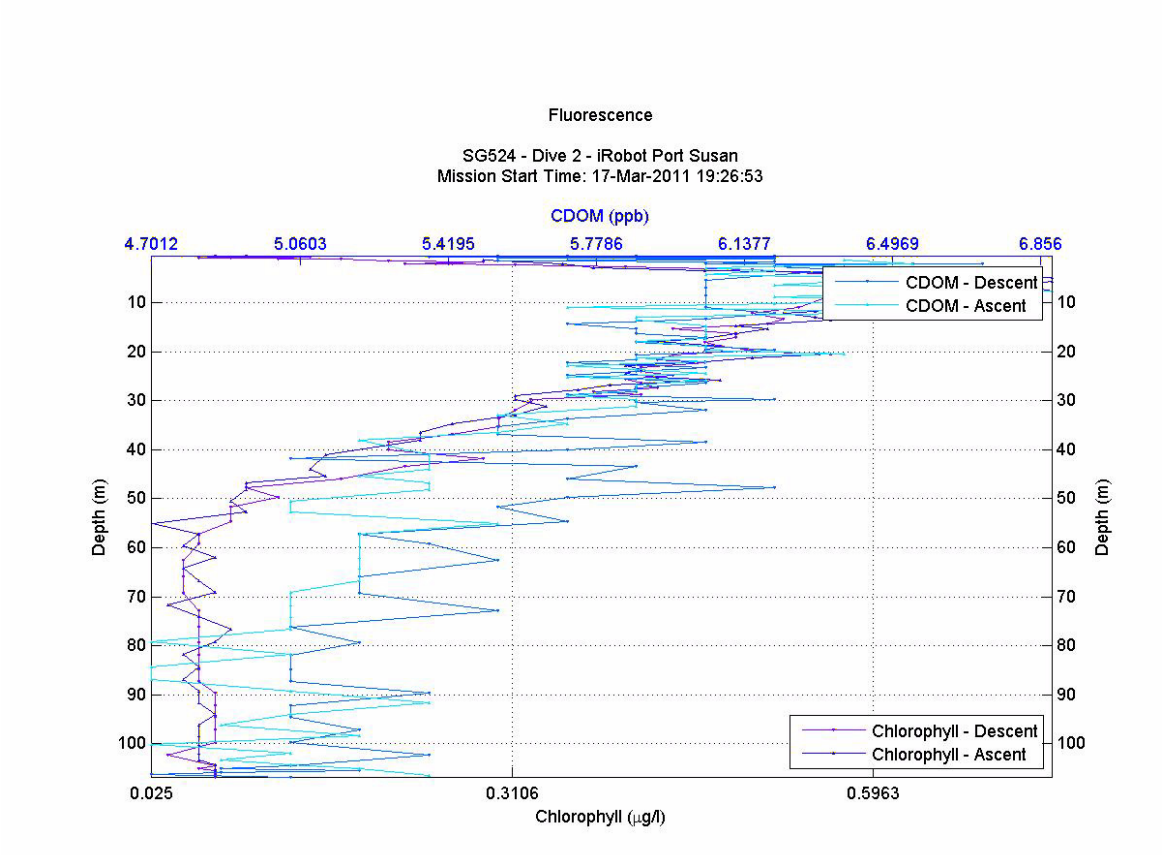
The backscatter plot displays measured wavelengths of signals from concentrations of material in the water. Depending on what triplet is sensor is installed, up to three backscatter measurements may be present. In this case the sensor provides optical backscattering measurements at 650nm excitation on both the descent and ascent. Depending on the triplet installed, more than one wavelength may be displayed. As seen below, the first triplet only has 1 backscatter in the triplet configuration, whereas, the second plot below has 2 backscatter in the triplet configuration.




Scattering
SG125 - Dive 5 - iRobot Port Susan
Mission Start Time: 15-Jun-2011 18:17:49
700 nm (RED) - Descent
470 nm (BLUE) - Descent
700 nm (RED) - Ascent
470 nm (BLUE) - Ascent
Depth (m)
0
5
10
15
20
25
30
0.889
1.3271
1.7652
2.2034
2.6415
3.0796
3.5178
3.9559
4.394
4.8321
5.2703
β(θc) (m-1 sr-1)
x 10-3

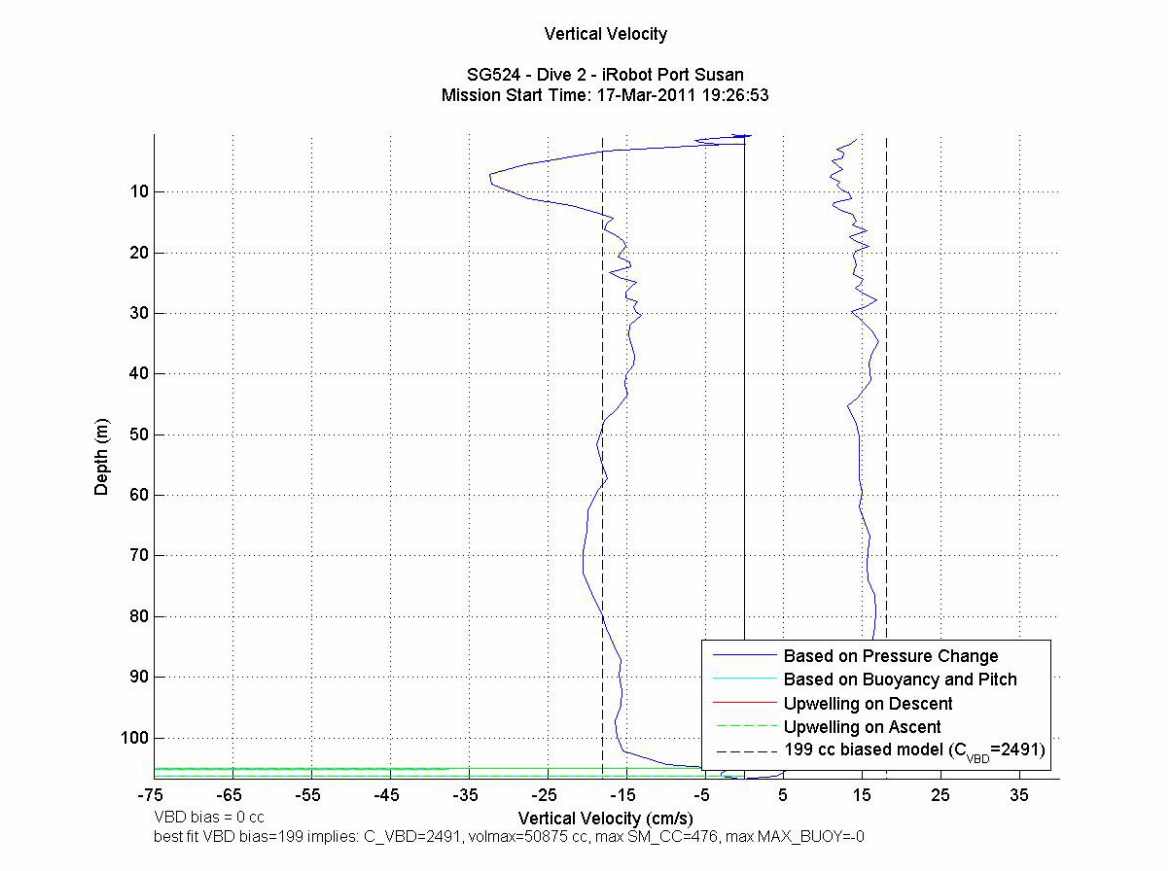
### Plot 6: Fluorometer Data (WET Labs Triplets)

The fluorometer plot displays the fluorescent response of substances in the water on the descent and ascent of the dive in different colors. In this case, CDOM and chlorophyll data are plotted.



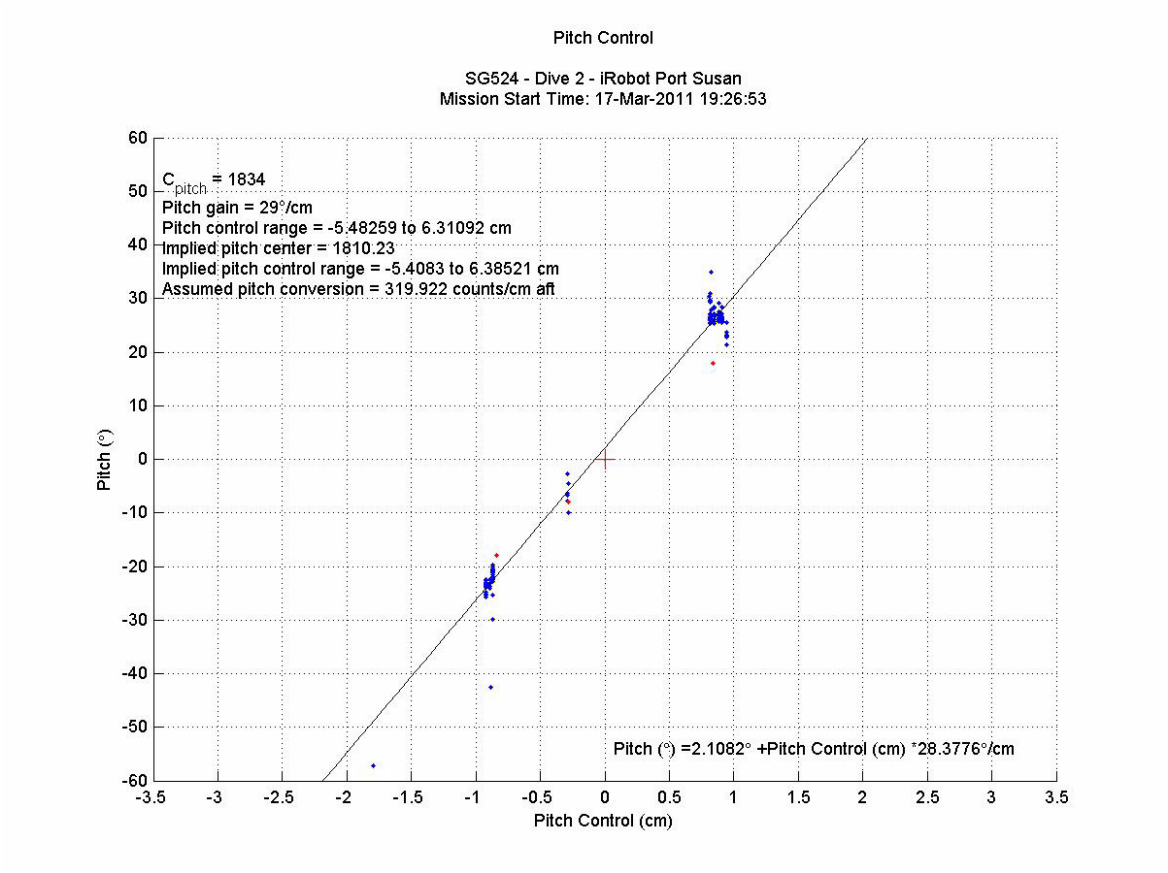
### Plot 7: Vertical Velocity

The vertical velocity plot measures vertical velocity (cm/s) on the dive's descent and ascent. The plot shows the vertical velocity of the glider based on 5 different methods of calculation. Ideally the pilot would like to see these centered around the same velocity throughout the entire dive. Ex: -15cm/s on dive and +15cm/s on the ascent.



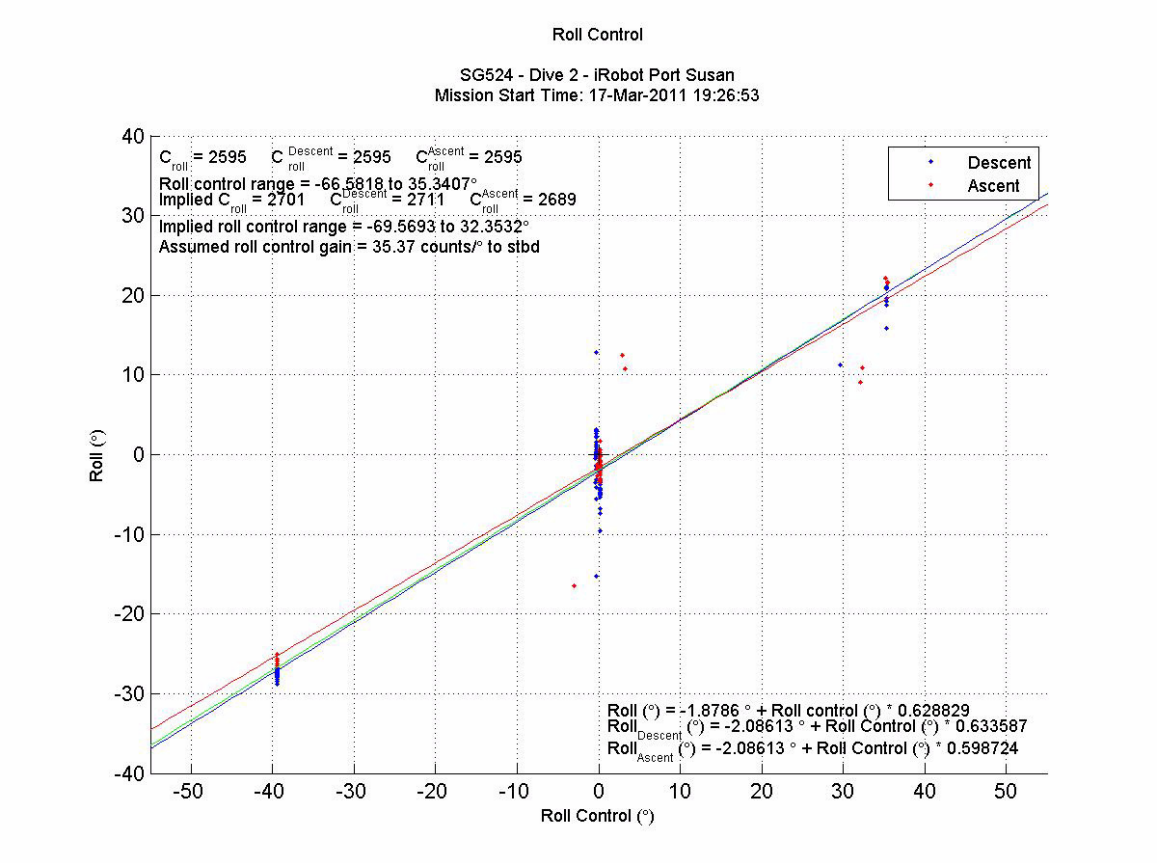
### Plot 8: Pitch Control Plot

The following pitch control plot displays the current pitch center and gain (C_PITCH, and PITCH_GAIN) parameters. Ideally the pilot would want the cm per degree regression line to cross through (0,0). The recommended pitch center and gain is also calculated and displayed on the plot. In the example below, changes are not necessary. The pitch center is off by 14 counts and the pitch gain is off by 0.6.


Pitch Control
SG524 - Dive 2 - iRobot Port Susan
Mission Start Time: 17-Mar-2011 19:26:53
C pitch = 1834
Pitch gain = 29°/cm
Pitch control range = -5.48259 to 6.31092 cm
Implied pitch center = 1810.23
Implied pitch control range = -5.4083 to 6.38521 cm
Assumed pitch conversion = 319.922 counts/cm aft
Pitch (°) =2.1082° +Pitch Control (cm) *28.3776°/cm
Pitch (°)
Pitch Control (cm)

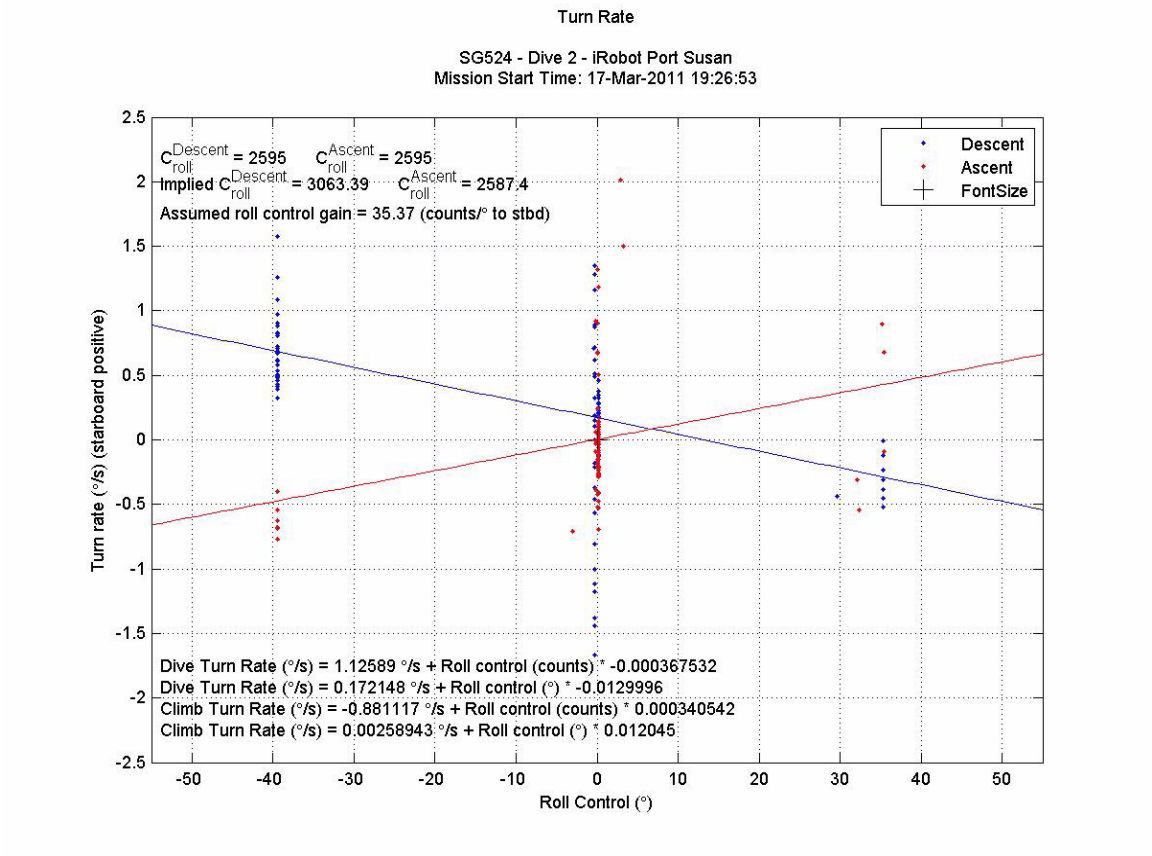
### Plot 9: Roll Control

Roll control is displayed on plot 9. The plot displays the roll control on the descent and ascent as well as the current roll centers and gain values. The roll center parameters are adjusted by pilots (C_ROLL_DIVE, C_ROLL_CLIMB); Ideally, both regression lines should pass through (0,0).



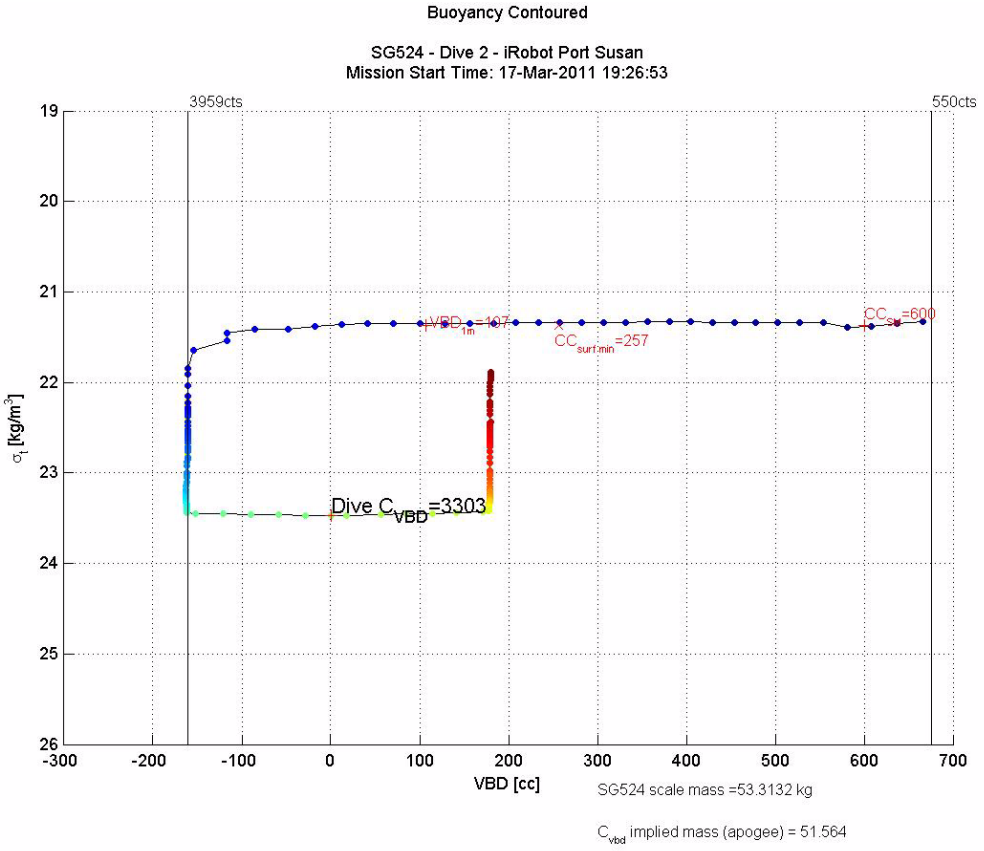
### Plot 10: Turn Rate Plot

The turn rate plot displays Turn Rate (deg/sec) vs. Roll Control (deg). Roll centers and gain are also displayed on this plot. This plot is used frequently in pilot control. Ideally, the descent and ascent should create an X pattern crossing at (0,0). The red and blue dots should be offset equally from the turn rate = 0 line. The red and blue lines should also run through the center of the red and blue dots respectively.



### Plot 11: Buoyancy Contoured Plot

The Buoyancy Plot displays VBD in cc's at different $\sigma_T$. The plot also displays the current C_VBD, SM_CC (surface maneuver cc's and the recommended cc's, $CC_{surfmin}$, needed at the surface. Implied mass at apogee is also displayed however this is not typically used while piloting. This plot allows the pilot to set optimal SM_CC for energy savings.

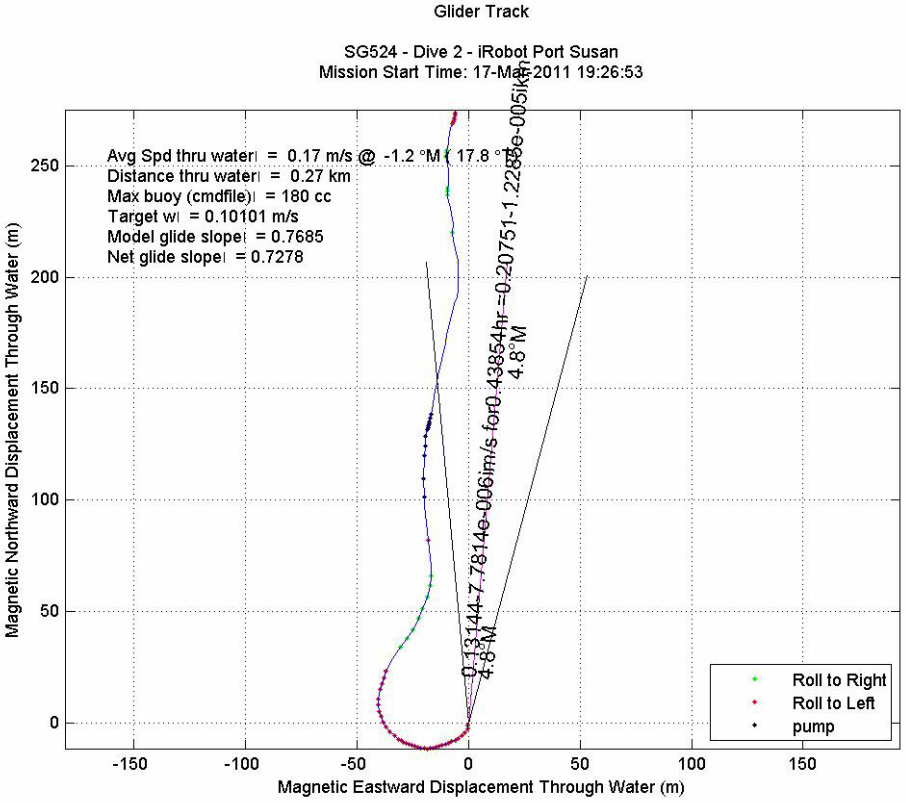


**Plot 12: Glider Track Plot**

The glider track plot displays the track and heading from an aerial view. Several flight characteristics can be viewed from this plot including the

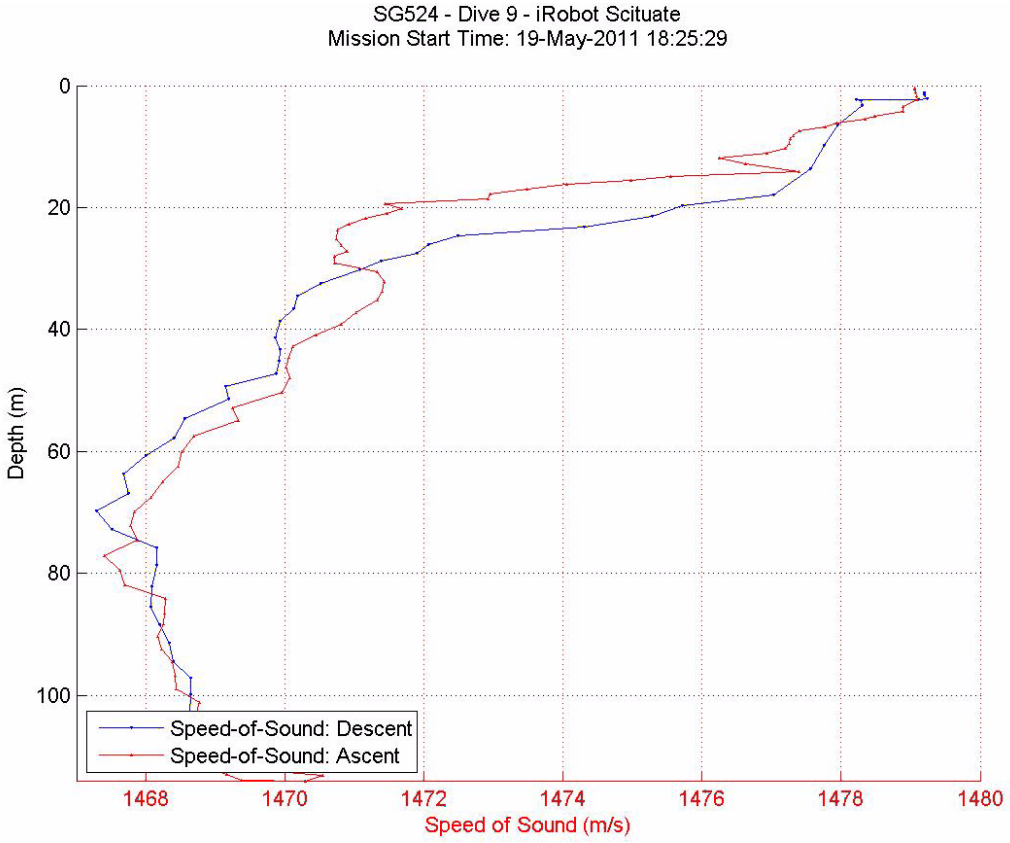
- Average Speed through water
- Distance through water
- Maximum Buoyancy (set by pilot)
- Current Target
- Glide Slopes

Also, seen on the plot is a ‘V’ which represents the heading error band. The actual track of the glider also shows roll maneuvers indicated by the red and green dots.


Glider Track
SG524 - Dive 2 - iRobot Port Susan
Mission Start Time: 17-Mar-2011 19:26:53
Avg Spd thru water = 0.17 m/s @ -1.2 °M ( 17.8 °
Distance thru water = 0.27 km
Max buoy (cmdfile) = 180 cc
Target w = 0.10101 m/s
Model glide slope = 0.7685
Net glide slope = 0.7278
0.13144+7.7814e-006i m/s for 0.43854hr =0.20751-1.2288e-005i km
4.8°M
4.8°M
Roll to Right
Roll to Left
pump
Magnetic Northward Displacement Through Water (m)
Magnetic Eastward Displacement Through Water (m)
250
200
150
100
50
0
-150
-100
-50
0
50
100
150

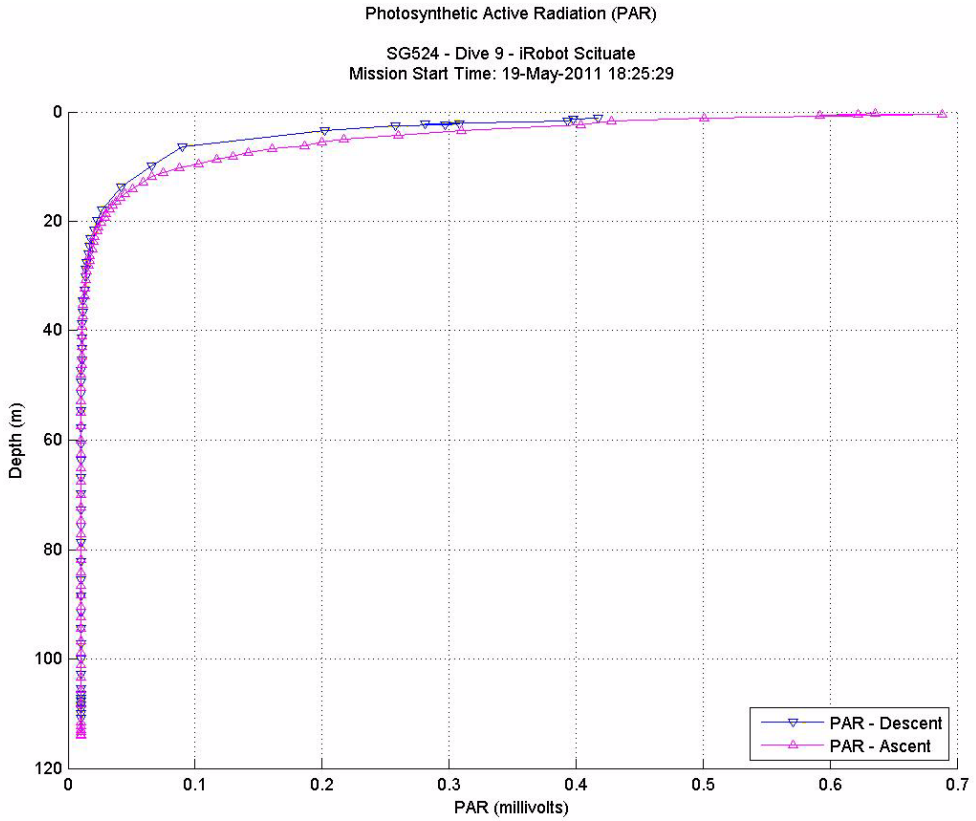
### Plot 13: Speed of Sound

The following plot displays the speed of sound in m/s at varying depths.



### Plot 14: PAR (Photosynthetically Active Radiation) Sensor Data

The PAR sensor measures the spectral range of solar radiation (400-700nm) used by aquatic plants and algae for photosynthesis. At deeper depths you will see the par sensor decrease to zero as solar radiation diminishes with depth.



APPENDIX E

# *Autonomous Self Test*

This appendix contains the autonomous self test.

```
------ Launch Menu ------
    1 [scene   ] Set scenario mode
    2 [selftest] Perform interactive self test
    3 [autotest] Perform autonomous self test
    4 [uploadst] Upload self-test results
    5 [reset   ] Reset dive/run number
    6 [test    ] Test Launch!
    7 [sea     ] Sea Launch!
   CR) Return to previous
Enter selection (1-7,CR): 3
60.347,SUSR,N,Beginning selftest #2 on glider SG535
0.627,SUSR,N,Tue Mar 29 10:37:26 2011
0.719,SUSR,N,---- Audible pings to mark start of tests ----
1.211,HXPDR,N,ping response: R008.926ms
1.272,HXPDR,N,range was 6.694500 m
3.626,HXPDR,N,ping response: R---------

3.688,HXPDR,N,range was 999.000000 m
5.776,HXPDR,N,ping response: R---------

5.838,HXPDR,N,range was 999.000000 m
7.778,SUSR,N,clearing tranponder ping count, prev count=0
7.898,SUSR,N,xpdr/ranging can be tested while other tests proceed
8.023,SUSR,N,---- Checking GPS ----
8.106,HGPS,N,Obtaining GPS fix
11.521,HGPS,N,satellite 2 almanac is -4 weeks old
11.716,HGPS,N,satellite 3 almanac is -4 weeks old
11.911,HGPS,N,satellite 4 almanac is -4 weeks old
12.107,HGPS,N,satellite 6 almanac is -5 weeks old
12.302,HGPS,N,satellite 7 almanac is -5 weeks old
12.497,HGPS,N,satellite 9 almanac is -5 weeks old
12.937,HGPS,N,satellite 11 almanac is -4 weeks old
13.133,HGPS,N,satellite 13 almanac is -5 weeks old
13.329,HGPS,N,satellite 14 almanac is -5 weeks old
13.527,HGPS,N,satellite 16 almanac is -4 weeks old
13.725,HGPS,N,satellite 21 almanac is -4 weeks old
13.922,HGPS,N,satellite 22 almanac is -4 weeks old
14.121,HGPS,N,satellite 24 almanac is -4 weeks old
14.317,HGPS,N,satellite 25 almanac is -4 weeks old
14.514,HGPS,N,satellite 26 almanac is -4 weeks old
14.711,HGPS,N,satellite 27 almanac is -4 weeks old
14.906,HGPS,N,satellite 31 almanac is -5 weeks old
14.980,HTT8,N,Updating parameter $T_GPS_CHARGE to -143.23141
17.387,HGPS,N,Acquiring GPS fix ...
18.478,HGPS,N,VVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVVA
3135283.001,HGPS,N,sync sentence
$GPRMC,173208,A,4142.6018,N,07046.4350,W,000.0,000.0,040511,014.9,W*7F
3135283.131,HGPS,N,set 2011/05/04 17:32:09
3135283.581,HGPS,N,confirm $GPRMC,173209,A,4142.6031,N,07046.4349,W,000.0,000.0,040511,014.9,W*7D
3135283.728,HGPS,N,040511, 173207, 4142.601074, -7046.435547 1.400000 35/35 seconds
3135284.343,HTT8,N,Updating parameter $T_GPS_CHARGE to -143.86217
3135287.030,SUSR,N,---- Checking Iridium phone ----
```

```
3135287.139,SUSR,N,---- Checking bathymetry data ----
3135287.251,SUSR,N,Normal Heap: 0+73727 bytes
3135287.389,SGLMALLOC,N,glheap_walk: 355020 bytes free, 2 blocks free, 2804 bytes alloc, 22 blocks alloc
3135290.222,SBATHY,N,Loaded bathymap.001 [101 100] 100m (LL) 47.8833,-122.4000 (UR) 47.9742,-122.2658
3135294.432,SBATHY,N,Loaded bathymap.003 [ 47 174] 100m (LL) 48.0333,-122.4667 (UR) 48.0756,-122.2325
3135300.237,SBATHY,N,Loaded bathymap.002 [102 112] 100m (LL) 47.9500,-122.3833 (UR) 48.0418,-122.2328
3135307.756,SBATHY,N,Loaded bathymap.004 [112 137] 100m (LL) 48.0667,-122.4833 (UR) 48.1675,-122.2988
3135308.099,SBATHY,N,WARNING: No bathymetry file covers the current location (4142.601074,-7046.435547)!
3135308.320,SGLMALLOC,N,glheap_walk: 324084 bytes free, 3 blocks free, 33668 bytes alloc, 24 blocks alloc
3135308.508,SUSR,N,---- Checking compass and calibration files ----
3135308.994,HCOMPASS,N,Compass 1, SP3003
3135309.244,HCOMPASS,N,pitch A,B,C,D: -0.015500 1.037100 -0.029100 -0.002100
3135309.550,HCOMPASS,N,roll A,B,C,D: -0.031000 -1.013800 2.108300 0.013000
3135309.841,HCOMPASS,N,hard iron P,Q,R: -14.898300 -19.541700 -23.372700
3135310.093,HCOMPASS,N,soft iron: 1.002600 -0.021200 -0.007200
3135310.329,HCOMPASS,N,          0.008700 1.060100 -0.009100
3135310.569,HCOMPASS,N,          0.015900 0.023800 1.069900
3135312.087,HCOMPASS,N,compass filter disabled ok
3135312.878,HCOMPASS,N,1, SP3003, Hdg: 102.31 deg Rol: -7.81 deg Pit: -73.76 deg Tmp: 23.50 C
3135314.318,SUSR,N,---- Reporting hardware configuration ----
3135314.459,SUSR,N,Sensor in slot 1 is WL_BBFL2VMT on port 3, TPU06/TPU07, nominally 'Optics 1'
3135314.629,SUSR,N,Sensor in slot 2 is not installed
3135314.740,SUSR,N,Sensor in slot 3 is not installed
3135314.851,SUSR,N,Sensor in slot 4 is not installed
3135314.962,SUSR,N,Sensor in slot 5 is not installed
3135315.073,SUSR,N,Sensor in slot 6 is not installed
3135315.209,SUSR,N,Logger sensor in logger slot 1 is GPCTD on port 5, TPU10/TPU11 (mux channel 0), nominally 'Optics 2'
3135315.414,SUSR,N,Logger Sensor in logger slot 2 is not installed
3135315.547,SUSR,N,Logger Sensor in logger slot 3 is not installed
3135315.680,SUSR,N,Logger Sensor in logger slot 4 is not installed
3135315.807,SUSR,N,Motherboard is Rev.B 1
3135315.898,SUSR,N,Phone is Iridium9522
3135315.986,SUSR,N,GPS is Garmin_GPS15H
3135316.075,SUSR,N,Compass is SP3003
3135316.161,SUSR,N,Spare compass is not installed
3135316.263,SUSR,N,XPDR is AAE_955
3135316.407,SUSR,N,External pressure sensor gain (128.000000) consistent with Paine
3135316.559,SUSR,N,---- Reporting software version ----
3135316.704,SUSR,N,Version: 66.07
3135316.788,SUSR,N,Compiled on: Mar 29 2011 10:36:24
3135316.896,SUSR,N,From SVN version 18:775
3135316.987,SUSR,N,Compiler: gcc
3135317.070,SUSR,N,Compiled with SCENARIO support
3135317.175,SUSR,N,Compiled without ICE support
3135317.277,SUSR,N,Compiled without RAFOS support
3135317.382,SUSR,N,Compiled without Kermit support
3135317.817,SUSR,N,Compiled without ARS support
3135317.920,SUSR,N,Compiled without PAAM support
3135318.024,SUSR,N,Compiled without AQUADOPP support
3135318.134,SUSR,N,Compiled without SAILCT support
3135318.241,SUSR,N,Compiled without MicroModem support
3135318.354,SUSR,N,Compiled without DeepGlider support
```

3135318.467,SUSR,N,Compiled without Heap Recycle support
3135318.584,SUSR,N,Compiled with Rev C Motherboard support
3135318.701,SUSR,N,Compiled without LUA support
3135318.814,SUSR,N,NVRAM: 178 parameters, 716 bytes, 186 bytes in utility storage
3135318.959,SUSR,N,Buffers: static
3135319.039,SUSR,N,Active (2Mhz)
3135319.121,SUSR,N,PicoDOS version: C82#2876-1.63b
3135319.225,SUSR,N,Launched as: MAIN
3135319.314,SUSR,N,Normal Heap: 0+73727 bytes
3135319.456,SGLMALLOC,N,glheap_walk: 324044 bytes free, 3 blocks free, 33684 bytes alloc, 25 blocks alloc
3135319.632,SUSR,N,Sensors: WL_BBFL2VMT
3135319.841,SUSR,N,MAIN.RUN: 490000 bytes

```
3135319.936,SUSR,N,---- Reporting directory contents ----
OLDMAIN.RUN         490020
GPCTD.CNF              699
BATHYMAP.001         33792
BATHYMAP.003         26752
BATHYMAP.002         38272
BATHYMAP.004         48896
SCIENCE                145
TARGETS                361
SG001PRM.TXT          2785
TCM2MAT.001            192
THISDIVE.KAP          6190
BATTERY                486
TCM53500.AVE          4851
TCM53500.RAW         85317
TCM2MAT.535            256
CURRENTS               220
GPCTDDO.CNF            803
MAIN.RUN            490000
QSP2150.CNF            128
WLBB2F.CNF             219
WLBB2FL.CNF            190
WLBBFL2.CNF            237
THISDIVE.LOG          4032
CMDFILE                989
THISDIVE.WRK          3821
THISDIVE.DAT           289
BG042011               465
LOGGERS.CMD            115
BG032911               465
RUN099.CFG             322
BG050311               482
BG042511               465
SG0000KZ.A            4545
BG042611               465
PDOSLOG.GZ             222
```

```
SG0099LZ.A       1667
SG0000LZ.A       1412
SG0001LZ.A       2677
BG042811          469
ST0001LU.A       4282
SG0099KZ.A       5737
SG0099DZ.A       1401
ST0001DU.A        932
PDOSLOG             0
SG0099DU.A       4224
SG0098KZ.A       1485
SG0098LZ.A       1428
PC0003AZ.X        371
PC0003BZ.X        364
ST0001KZ.A      21568
SG0001DZ.A       1278
SG0001KZ.A       7917
SG0002KZ.A       8055
SG0002DZ.A       1324
SG0002LZ.A       2709
1316788 bytes used; 55 files
3135322.352,SUSR,N,---- Checking pressure sensor ----
3135323.656,HPRES,N,                A/D         Volts     Meters
3135323.909,HPRES,N, Mean:          458189.8    0.532336  -0.08
3135324.167,HPRES,N, RMS:           458190.0    0.532337  -0.08
3135324.381,HPRES,N, Min:           457391      0.531408  -0.15
3135324.588,HPRES,N, Max:           459317      0.533646  0.01
3135324.766,HPRES,N, Peak-to-peak:  1926        2.237675  uV

3135324.933,HPRES,N, Volts per bit: 1.161825e-09

3135325.097,HPRES,N,Current pressure y-intercept is -53.262634 (psig).
3135325.290,HPRES,N,If truly at sealevel, this data suggests it should be -53.140045 (psig).
3135325.506,HTT8,N,Updating parameter $PRESSURE_YINT to -53.140045
3135325.840,HTT8,N,Writing NVRAM...done.
3135333.988,HPRES,N,new value automatically accepted
3135334.855,HPRES,N,                A/D         Volts     Meters
3135335.104,HPRES,N, Mean:          457038.1    0.530998  -0.09
3135335.376,HPRES,N, RMS:           457038.2    0.530998  -0.09
3135335.590,HPRES,N, Min:           456387      0.530242  -0.14
3135335.797,HPRES,N, Max:           458287      0.532449  0.01
3135335.977,HPRES,N, Peak-to-peak:  1900        2.207467  uV
3135336.144,HPRES,N, Volts per bit: 1.161825e-09

3135336.308,HPRES,N,Current pressure y-intercept is -53.140045 (psig).
3135336.499,HPRES,N,If truly at sealevel, this data suggests it should be -53.006554 (psig).
```

```
3135336.712,HTT8,N,Updating parameter $PRESSURE_YINT to -53.006554
3135337.047,HTT8,N,Writing NVRAM...done.
3135345.195,HPRES,N,new value automatically accepted
3135346.654,HPRES,N,                A/D         Volts     Meters
3135346.902,HPRES,N, Mean:          457472.3   0.531503   0.03
3135347.170,HPRES,N, RMS:           457472.2   0.531503   0.03
3135347.376,HPRES,N, Min:           457111     0.531083   0.01
3135347.586,HPRES,N, Max:           457944     0.532051   0.07
3135347.763,HPRES,N, Peak-to-peak:  833        0.967800   uV

3135347.931,HPRES,N, Volts per bit: 1.161825e-09

3135348.037,HPRES,N,Y-intercept is set correctly
3135348.159,SUSR,N,---- Checking GPCTD ----
3135348.252,HGPCTD,N,syncing clock
clock sync string is %r%[S>]RESETLOGGING%r%[confirm%n]RESETLOGGING%r%[S>]%{DATETIME=%m%d%Y%H%M%S}%[S>]
fmt into strftime = [DATETIME=%m%d%Y%H%M%S]
3135354.974,HGPCTD,N,sending [DATETIME=05042011173323]sleeping 80762 ticks, sync will be 0x0d
3135360.260,HGPCTD,N,sample: TS
003EA455480138C04219
S>---- Checking pitch, roll, and VBD. ----
3135360.450,SUSR,N,---- Checking pitch motor ----
3135360.552,SUSR,N,Pitching up...
3135360.762,HPITCH,N,Pitch commanded from -4.69 cm (89) to 7.15 cm (3879)...
3135361.723,HPITCH,N,-4.4 cm (ad: 197) ...................
3135372.473,HPITCH,N,4.3 cm (ad: 2981) .......MOTOR_DONE: ticks: 14 max 24v: 3.8mA avg 24v: 3.1mA minV 24v: 27.0V
3135376.712,SMOTOR,N,GC TICKS/TIME: 607/609145
3135376.808,HPITCH,N,done.
3135376.954,HPITCH,N,Pitch completed from -4.69 cm (89) to 7.12 cm (3868) took 14.8 sec 236mA (644mA peak) 26.5Vmin 256 AD/sec 590 ticks
3135377.369,HPITCH,N,TRACK: b: 3863/0 a: 3866/0 d: 0 o: 2
3135377.482,SUSR,N,Pitching down...
3135377.695,HPITCH,N,Pitch commanded from 7.11 cm (3867) to -4.61 cm (115)...
3135378.658,HPITCH,N,6.6 cm (ad: 3708) ..................MOTOR_DONE: ticks: 3 max 24v: 3.1mA avg 24v: 3.1mA minV 24v: 26.9V
3135388.646,SMOTOR,N,GC TICKS/TIME: 407/409196
3135388.742,HPITCH,N,done.
3135388.889,HPITCH,N,Pitch completed from 7.11 cm (3867) to -4.61 cm (115) took 10.0 sec 62mA (285mA peak)
26.8Vmin 374 AD/sec 401 ticks
3135389.293,HPITCH,N,TRACK: b: 131/0 a: 127/0 d: 0 o: -12
3135389.408,SUSR,N,Backing off pitch mass...
3135389.621,HPITCH,N,   Pitch control position = -4.62 cm, (AD#= 114 )
3135389.884,HPITCH,N,Pitch commanded from -4.62 cm (114) to -2.62 cm (753)...
3135390.876,HPITCH,N,-4.3 cm (ad: 219) ....MOTOR_DONE: ticks: 8 max 24v: 3.1mA avg 24v: 3.1mA minV 24v: 26.9V
3135393.492,SMOTOR,N,GC TICKS/TIME: 112/114262
3135393.587,HPITCH,N,done.
3135393.734,HPITCH,N,Pitch completed from -4.62 cm (114) to -2.65 cm (742) took 2.5 sec 243mA (296mA peak) 26.5Vmin 248 AD/sec 101 ticks
3135394.137,HPITCH,N,TRACK: b: 737/0 a: 740/0 d: 0 o: 2
```

```
3135394.250,SUSR,N,---- Checking roll motor ----
3135394.350,SUSR,N,Roll to port...
3135394.566,HROLL,N,Roll commanded from -0.11 deg (2800) to 27.93 deg (3792)...
3135395.531,HROLL,N,6.4 deg (ad: 3032) ..MOTOR_DONE: ticks: 3 max 24v: 3.1mA avg 24v: 3.1mA minV 24v: 26.9V
3135397.118,SMOTOR,N,GC TICKS/TIME: 71/73178
3135397.211,HROLL,N,done.
3135397.361,HROLL,N,Roll completed from -0.11 deg (2800) to 27.82 deg (3788) took 1.6 sec 70mA (251mA peak) 26.7Vmin 608 AD/sec 65 ticks
3135397.756,HROLL,N,TRACK: b: 3765/0 a: 3769/0 d: -1 o: 19
3135397.872,SUSR,N,Roll to starboard...
3135398.097,HROLL,N,Roll commanded from 27.82 deg (3788) to -40.00 deg (1389)...
3135399.090,HROLL,N,21.4 deg (ad: 3561) .......MOTOR_DONE: ticks: 18 max 24v: 3.8mA avg 24v: 3.1mA minV 24v: 27.0V
3135403.279,SMOTOR,N,GC TICKS/TIME: 176/178090
3135403.374,HROLL,N,done.
3135403.530,HROLL,N,Roll completed from 27.82 deg (3788) to -40.06 deg (1387) took 3.9 sec 63mA (233mA peak)
26.7Vmin 619 AD/sec 155 ticks
3135403.932,HROLL,N,TRACK: b: 1415/0 a: 1410/0 d: 1 o: -23
3135404.050,SUSR,N,Roll to neutral...
3135404.240,HROLL,N,Roll commanded from -40.06 deg (1387) to 0.00 deg (2804)...
3135405.206,HROLL,N,-33.4 deg (ad: 1621) ....Updating parameter $R_STBD_OVSHOOT to 19
3135407.850,SMOTOR,N,MOTOR_DONE: ticks: 18 max 24v: 3.8mA avg 24v: 3.1mA minV 24v: 27.0V
3135408.008,SMOTOR,N,GC TICKS/TIME: 113/115312
3135408.104,HROLL,N,done.
3135408.255,HROLL,N,Roll completed from -40.06 deg (1387) to -0.14 deg (2799) took 2.3 sec 73mA (228mA peak) 26.7Vmin 613 AD/sec 92 ticks
3135408.653,HROLL,N,TRACK: b: 2773/0 a: 2780/0 d: 0 o: 19
3135408.770,SUSR,N,---- Checking VBD and valve ----
3135408.935,HVBD,N,   VBD lin pot AD counts: 3419 3277
3135409.089,HVBD,N,   VBD bladder position = -93.70 cc, (AD#= 3348 )
3135409.279,HVBD,N,   VBD lin pot AD counts: 3419 3277
3135409.433,HVBD,N,   VBD bladder position = -93.70 cc, (AD#= 3348 )
3135409.566,SUSR,N,Pumping 100cc on boost alone...
3135409.802,HVBD,N,Pump commanded from -93.46 cc (3347) to 6.38 cc (2940)...
3135410.764,HVBD,N,-92.5 cc (ad: 3343 [3417, 3267]) ...................
3135421.564,HVBD,N,-60.3 cc (ad: 3212 [3284, 3137]) ...................
3135432.364,HVBD,N,-27.7 cc (ad: 3079 [3145, 3010]) ...................
3135443.164,HVBD,N,4.2 cc (ad: 2949 [3005, 2889]) .MOTOR_DONE: ticks: 4 max 24v: 4.6mA avg 24v: 4.6mA minV 24v: 26.8V
3135444.256,SMOTOR,N,GC TICKS/TIME: 1347/1349244
3135444.354,HVBD,N,done.
3135444.505,HVBD,N,Pump completed from -93.70 cc (3348) to 6.87 cc (2938 [2996, 2880]) took 33.5 sec 134mA
(1535mA peak) 26.5Vmin 12 AD/sec 1340 ticks
3135444.951,HVBD,N,TRACK: b: 2999/2882 a: 2999/2882 d: -59 o: -61
3135445.078,SUSR,N,Pumping up to maximum buoyancy...
3135445.320,HVBD,N,Pump commanded from 7.60 cc (2935) to 607.11 cc (491)...
3135446.284,HVBD,N,8.6 cc (ad: 2931 [2986, 2872]) ...................
3135457.034,HVBD,N,39.7 cc (ad: 2804 [2865, 2739]) ...................
3135467.834,HVBD,N,70.6 cc (ad: 2678 [2728, 2625]) ...................
3135478.634,HVBD,N,101.1 cc (ad: 2554 [2596, 2508]) ...................
3135489.409,HVBD,N,131.5 cc (ad: 2430 [2476, 2381]) ...................
```

```
3135500.184,HVBD,N,161.4 cc (ad: 2308 [2353, 2260]) ...................
3135510.959,HVBD,N,191.1 cc (ad: 2187 [2224, 2146]) ...................
3135521.734,HVBD,N,220.3 cc (ad: 2068 [2104, 2029]) ...................
3135532.534,HVBD,N,250.0 cc (ad: 1947 [1981, 1911]) ...................
3135543.334,HVBD,N,279.1 cc (ad: 1828 [1860, 1793]) ...................
3135554.134,HVBD,N,308.3 cc (ad: 1709 [1741, 1675]) ...................
3135564.909,HVBD,N,337.3 cc (ad: 1591 [1621, 1559]) ...................
3135575.734,HVBD,N,365.7 cc (ad: 1475 [1499, 1448]) ...................
3135586.509,HVBD,N,394.2 cc (ad: 1359 [1382, 1334]) ...................
3135597.309,HVBD,N,422.6 cc (ad: 1243 [1264, 1222]) ...................
3135608.084,HVBD,N,450.6 cc (ad: 1129 [1145, 1110]) ...................
3135618.884,HVBD,N,478.6 cc (ad: 1015 [1028, 997]) ...................
3135629.659,HVBD,N,506.5 cc (ad: 901 [917, 883]) ...................
3135640.434,HVBD,N,534.7 cc (ad: 786 [802, 768]) ...................
3135651.209,HVBD,N,562.2 cc (ad: 674 [690, 657]) ...................
3135661.984,HVBD,N,589.7 cc (ad: 562 [579, 543]) .............MOTOR_DONE: ticks: 14 max 24v: 6.9mA avg 24v: 4.6mA
minV 24v: 25.7V
3135669.393,SMOTOR,N,GC TICKS/TIME: 8932/8933973
3135669.492,HVBD,N,done.
3135669.644,HVBD,N,Pump completed from 7.36 cc (2936) to 607.84 cc (488 [507, 469]) took 222.9 sec 568mA (1577mA
peak) 25.1Vmin 10 AD/sec 8915 ticks
3135670.087,HVBD,N,TRACK: b: 509/474 a: 509/474 d: -18 o: -21
0.241,SSURF,N,Initializing data file
0.476,SSURF,N,Initializing log file
2.284,SUSR,N,---- Checking sensors and data file creation ----
0.436,SSURF,N,Dive started Wed May  4 17:38:38 2011
 (1304530718)
5.603,SUSR,N,entering SD active
9.483,SSENSOR,N,A 5320ms 0.03m 101.2 #1
9.778,SUSR,N,entering SD passive
13.412,SSENSOR,N,P 10091ms 0.07m 101.7 #2
14.399,SUSR,N,entering SD active
18.282,SSENSOR,N,A 14116ms 0.07m 101.1 #3
18.593,SUSR,N,entering SD passive
22.235,SSENSOR,N,P 18908ms 0.16m 101.3 #4
23.214,SUSR,N,entering SD active
26.862,SSENSOR,N,A 22932ms 0.09m 101.1 #5
27.186,SUSR,N,entering SD passive
31.506,SSENSOR,N,P 27897ms 0.03m 101.7 #6
32.389,SUSR,N,entering SD active
36.042,SSENSOR,N,A 32106ms 0.15m 101.3 #7
36.377,SUSR,N,entering SD passive
40.235,SSENSOR,N,P 36692ms 0.00m 101.1 #8
41.131,SUSR,N,entering SD active
44.781,SSENSOR,N,A 40849ms 0.10m 101.8 #9
45.128,SUSR,N,entering SD passive
```

```
48.990,SSENSOR,N,P 45442ms 0.05m 101.5 #10
49.900,SUSR,N,entering SD active
53.546,SSENSOR,N,A 49617ms 0.04m 101.5 #11
53.907,SUSR,N,entering SD passive
57.769,SSENSOR,N,P 54221ms -0.01m 101.5 #12
58.690,SUSR,N,entering SD active
62.343,SSENSOR,N,A 58407ms 0.16m 101.5 #13
62.715,SUSR,N,entering SD passive
66.577,SSENSOR,N,P 63029ms 0.08m 101.5 #14
67.513,SUSR,N,entering SD active
71.164,SSENSOR,N,A 67230ms 0.10m 101.3 #15
version: 66.07
glider: 535
mission: 0
selftest: 2
start: 5 4 111 17 38 36
columns:
rec,elaps_tms,depth,heading,pitch,roll,AD_pitch,AD_roll,AD_vbd,GC_phase,wlbbfl2vmt.600ref,wlbbfl2vmt.600sig,wlbb
fl2vmt.Chlref,wlbbfl2vmt.Chlsig,wlbbfl2vmt.Cdomref,wlbbfl2vm.Cdomsig,wlbbfl2vmt.L2VMTtemp,
data:
0 5320 2 1011 -200 -67 725 2799 488 6 650 4130 695 118 460 155 539
1 10091 6 1016 -200 -70 725 2799 488 6 650 4130 695 124 460 177 539
2 14116 6 1011 -200 -66 725 2799 488 6 650 4130 695 107 460 170 539
3 18908 16 1012-200 -66 725 2799 488 6 650 4130 695 125 460 162 539
4 22932 9 1010 -200 -65 725 2799 488 6 650 4130 695 119 460 177 539
5 27897 2 1017 -200 -70 725 2799 488 6 650 4130 695 116 460 159 539
6 32106 15 1013 -200 -68 725 2799 488 6 650 4130 695 125 460 171 59
7 36692 0 1011 -200 -64 725 2799 488 6 650 4130 695 112 460 162 539
8 40849 10 1018 -200 -72 725 2799 488 6 650 4130 695 127 460 170 538
9 45442 4 1014 -200 -68 725 2799 488 6 650 4130 695 105 460 165 538
10 49617 3 1014 -200 -70 725 2799 488 6 650 4130695 108 460 160 538
11 54221 -1 1014 -200 -69 725 2799 488 6 650 4130 695 102 460 168 539
12 58407 16 1015 -200 -68 725 2799 488 6 650 4130 695 126 460 169 538
13 63029 8 1014 -200 -70 725 2799 488 6 650 4130 695 111 460 162 538
14 67230 10 1012 -200 -68 25 2799 488 6 650 4130 695 116 460 159 538
75.690,SUSR,N,---- Writing internal sensors to log ----
76.629,SUSR,N,internal humidity 1632 counts (38.457661RH)
76.838,SUSR,N,internal pressure 9.339511 psia
76.938,SUSR,N,---- Checking altimeter ----
77.404,HXPDR,N,ping response: R005.031ms
77.467,HXPDR,N,range was 3.773250 m
77.827,SUSR,N,---- Checking transponder ----
78.307,HXPDR,N,Tried 1 times to respond to command $C1
```

```
78.556,SUSR,N,transponder reply ping count=2
78.860,HTT8,N,Writing NVRAM...done.
87.006,SUSR,N,---- Reporting targets and science specifications ----
89.716,SNAV,N,       Target     Latitude  Longitude    Radius Depth Finish Timeout Next target Escp target
90.062,SNAV,N,   =>  SEVEN      4807.000  -12223.000    200.0m    0  -1   0.0 SIX none
90.379,SNAV,N,       SIX        4806.000  -12222.000    200.0m    0  -1   0.0 FIVE none
90.699,SNAV,N,       FOUR       4804.000  -12220.000    200.0m    0  -1   0.0 FIVE none
91.018,SNAV,N,       EIGHT      4808.000  -12224.000    200.0m    0  -1   0.0 KAYAKPT none
91.342,SNAV,N,       KAYAKPT    4808.000  -12223.000    100.0m    0  -1   0.0 KAYAKPT none
91.665,SNAV,N,       FIVE       4805.000  -12221.000    200.0m    0  -1   0.0 EIGHT none
92.115,SSENSOR,N,     Depth      Time       G&C Sensors
92.306,SSENSOR,N,     15.0m      5.0s     60.0s 111 (WL_BBFL2VMT: 1)
92.558,SSENSOR,N,     50.0m      5.0s    120.0s 555 (WL_BBFL2VMT: 5)
92.813,SSENSOR,N,    150.0m      5.0s    120.0s 777 (WL_BBFL2VMT: 7)
92.979,SUSR,N,---- Reporting battery status ----

97.321,SPOWER,N,------ Fuel gauges (Cumulative Amp-secs) -------

                         cumulative      / since power up
97.606,SPOWER,N,              Pitch_motor:    32.36 amp-sec /    27.30 sec
97.809,SPOWER,N,               Roll_motor:     4.03 amp-sec /     7.80 sec
97.997,SPOWER,N,     VBD_pump_during_apogee:   169.29 amp-sec /     0.00 sec
98.229,SPOWER,N,     VBD_pump_during_surface:   999.05 amp-sec /   256.38 sec
98.386,SPOWER,N,                VBD_valve:     0.00 amp-sec /     0.00 sec
98.571,SPOWER,N,      Iridium_during_init:   118.37 amp-sec /     0.00 sec
98.759,SPOWER,N,     Iridium_during_connect:   289.37 amp-sec /     0.00 sec
99.279,SPOWER,N,      Iridium_during_xfer:   676.99 amp-sec /     0.00 sec
99.473,SPOWER,N,         Transponder_ping:     2.52 amp-sec /     1.50 sec
99.626,SPOWER,N,               GUMSTIX_24V:     0.00 amp-sec /     0.00 sec
99.802,SPOWER,N,       ------------- 24V total = 0.637 AmpHr -----------
100.008,SPOWER,N,                    GPS:     7.38 amp-sec /    45.81 sec
100.230,SPOWER,N,                    TT8:   181.22 amp-sec /   155.71 sec
100.437,SPOWER,N,                 LPSleep:     4.91 amp-sec /     4.99 sec
100.658,SPOWER,N,              TT8_Active:    43.93 amp-sec /   295.45 sec
100.884,SPOWER,N,            TT8_Sampling:   447.73 amp-sec /   114.41 sec
101.101,SPOWER,N,                 TT8_CF8:    23.55 amp-sec /    32.45 sec
101.281,SPOWER,N,              TT8_Kalman:     5.37 amp-sec /     0.00 sec
101.503,SPOWER,N,         Analog_circuits:    48.60 amp-sec /   361.65 sec
101.657,SPOWER,N,            GPS_charging:     0.00 amp-sec /     0.00 sec
101.872,SPOWER,N,                 Compass:    41.04 amp-sec /    55.32 sec
102.026,SPOWER,N,                   RAFOS:     0.00 amp-sec /     0.00 sec
102.227,SPOWER,N,             Transponder:     0.94 amp-sec /     7.03 sec
102.381,SPOWER,N,                Compass2:     0.00 amp-sec /     0.00 sec
102.599,SPOWER,N,              WL_BBFL2VMT:   126.16 amp-sec /    49.68 sec
102.820,SPOWER,N,                   GPCTD:   437.60 amp-sec /    11.83 sec
103.003,SPOWER,N,       ------------- 10V total = 0.380 AmpHr ------------
```

```
103.361,HBATT,N,24V batt pack voltage = 26.32V
103.497,HBATT,N,10V batt pack voltage = 11.06V
106.900,SUSR,N,---- Checking capture vector ----
107.008,SUSR,N,all capture settings at default values
107.117,SUSR,N,---- Checking flash ----
No flash problems detected
118.275,SUSR,N,*** EOTD ***


119.140,SUSR,N,---- Checking communications ----
119.250,SUSR,N,Testing comms by transmitting self-test results:
119.749,HCF8,N,file 'pc0003az.x00' opened...


121.049,HCF8,N,file 'pc0003bz.x00' opened...


122.225,HCF8,N,file 'st0002lu.x00' opened...
122.374,HCF8,N,file 'st0002lu.x00' has 4 KB, closed...

123.141,HCF8,N,file 'st0002lu.x01' opened...


124.456,HCF8,N,file 'st0002du.x00' opened...


125.470,SGLMALLOC,N,glheap_walk: 322832 bytes free, 8 blocks free, 34176 bytes alloc, 50 blocks alloc
125.598,SSURF,N,Trying call 0...
125.642,SSURF,N,Calling phone number: 19193612847
129.762,HPHONE,N,initializing PSTN connection
151.062,HPHONE,N,Iridium signal strength: 5
151.227,HPHONE,N,Iridium geolocation: 41.441650 -70.759834 Tue May  3 20:50:08 2011
231.357,SSURF,N,no login: prompt detected [
CONNECT 19200
NO CARRIER
]
231.461,SSURF,N,Unable to login...
231.510,SPOWER,N,powerOFF (7,Iridium_during_xfer) without corresponding powerON!
231.614,SSURF,N,Total NAK's: 0, timeouts: 0
231.670,SSURF,N,going to sleep..
Type tt8:
245.085,SGLMALLOC,N,glheap_walk: 322832 bytes free, 8 blocks free, 34176 bytes alloc, 50 blocks alloc
245.213,SSURF,N,Trying call 1...
245.256,SSURF,N,Calling phone number: 19193612847
247.369,HPHONE,N,initializing PSTN connection
```

```
277.972,HPHONE,N,registration check failed - signal strength not checked
278.175,HPHONE,N,Iridium geolocation: 41.489616 -70.789391 Wed May  4 17:41:18 2011
306.444,SSURF,N,Unable to login...
306.492,SPOWER,N,powerOFF (7,Iridium_during_xfer) without corresponding powerON!
306.596,SSURF,N,Total NAK's: 0, timeouts: 0
306.653,SSURF,N,going to sleep..
Type tt8:
311.045,SGLMALLOC,N,glheap_walk: 322832 bytes free, 8 blocks free, 34176 bytes alloc, 50 blocks alloc
311.173,SSURF,N,Trying call 2...
311.217,SSURF,N,Calling phone number: 19193612847
313.328,HPHONE,N,initializing PSTN connection
345.277,HPHONE,N,Iridium signal strength: 5
345.442,HPHONE,N,Iridium geolocation: 41.489616 -70.789391 Wed May  4 17:41:18 2011
399.671,SSURF,N,Logged in...
401.381,SSURF,N,Sending cmd lsx -k -t200 cmdfile
403.244,SSURF,N,Receiving cmdfile...
404.700,SSURF,N,SOH/STX: 0x2 received (3), secSize= 1024...
409.380,SSURF,N,EOT: 0x4 received (1), secSize= 1024...
410.032,SSURF,N,Received cmdfile 1021 bytes
410.998,SSURF,N,Transmission succeeded...
411.085,HTT8,N,Updating parameter $CAPUPLOAD to 1
411.158,HTT8,N,Updating parameter $C_PITCH to 1591
411.424,HTT8,N,Updating parameter $C_ROLL_CLIMB to 2804
411.505,HTT8,N,Updating parameter $C_ROLL_DIVE to 2804
411.585,HTT8,N,Updating parameter $C_VBD to 2966
411.653,HTT8,N,Updating parameter $D_TGT to 30
411.722,HTT8,N,Updating parameter $MASS to 53188
411.792,HTT8,N,Updating parameter $MAX_BUOY to 150
411.864,HTT8,N,Updating parameter $PITCH_GAIN to 30
411.938,HTT8,N,Updating parameter $RHO to 1.026
412.006,HTT8,N,Updating parameter $SM_CC to 500
412.073,HTT8,N,Updating parameter $T_DIVE to 10
412.141,HTT8,N,Updating parameter $T_MISSION to 15
412.215,HTT8,N,Updating parameter $AH0_10V to 100
412.287,HTT8,N,Updating parameter $AH0_24V to 150
412.360,HTT8,N,Updating parameter $ALTIM_PING_DELTA to 5
412.441,HTT8,N,Updating parameter $ALTIM_PING_DEPTH to 80
412.526,HTT8,N,Updating parameter $ALTIM_BOTTOM_TURN_MARGIN to 12
412.614,HTT8,N,Updating parameter $D_NO_BLEED to 200
412.686,HTT8,N,Updating parameter $D_SURF to 3
412.749,HTT8,N,Updating parameter $KALMAN_USE to 0
412.820,HTT8,N,Updating parameter $NAV_MODE to 1
412.891,HTT8,N,Updating parameter $PITCH_AD_RATE to 175
412.972,HTT8,N,Updating parameter $PITCH_CNV to 0.003125763
413.058,HTT8,N,Updating parameter $PITCH_MAX to 3879
413.135,HTT8,N,Updating parameter $PITCH_MIN to 115
```

```
413.211,HTT8,N,Updating parameter $PITCH_TIMEOUT to 16
413.292,HTT8,N,Updating parameter $PRESSURE_SLOPE to 0.00011597844
413.385,HTT8,N,Updating parameter $ROLL_AD_RATE to 350
413.465,HTT8,N,Updating parameter $ROLL_CNV to 0.028270001
413.550,HTT8,N,Updating parameter $ROLL_MAX to 3792
413.626,HTT8,N,Updating parameter $ROLL_MIN to 231
413.703,HTT8,N,Updating parameter $SEABIRD_C_G to -11
413.785,HTT8,N,Updating parameter $SEABIRD_C_H to 0.60000002
413.875,HTT8,N,Updating parameter $SEABIRD_C_I to -0.0014
413.963,HTT8,N,Updating parameter $SEABIRD_C_J to 9.9999997e-05
414.056,HTT8,N,Updating parameter $SEABIRD_T_G to 0.0040000002
414.147,HTT8,N,Updating parameter $SEABIRD_T_H to 0.00060000003
414.241,HTT8,N,Updating parameter $SEABIRD_T_I to 2.4000001e-05
414.334,HTT8,N,Updating parameter $SEABIRD_T_J to 2.6e-06
414.666,HTT8,N,Updating parameter $T_GPS to 15
414.734,HTT8,N,Updating parameter $T_NO_W to 120
414.804,HTT8,N,Updating parameter $T_RSLEEP to 3
414.873,HTT8,N,Updating parameter $T_TURN_SAMPINT to 5
414.948,HTT8,N,Updating parameter $USE_BATHY to -4
415.024,HTT8,N,Updating parameter $VBD_CNV to -0.245296
415.106,HTT8,N,Updating parameter $VBD_MAX to 3961
415.181,HTT8,N,Updating parameter $VBD_MIN to 491
415.253,HTT8,N,Updating parameter $VBD_PUMP_AD_RATE_SURFACE to 5
415.345,HTT8,N,Updating parameter $VBD_TIMEOUT to 720
415.422,HTT8,N,Updating parameter $XPDR_VALID to 2
416.058,HTT8,N,Updating parameter $LOGGERS to 1
416.125,SDIVE,N,Parsed command: $RESUME
416.420,HTT8,N,Writing NVRAM...done.
424.035,SUSR,N,$ID,535
424.068,SUSR,N,$MISSION,0
424.103,SUSR,N,$DIVE,0
424.138,SUSR,N,$D_SURF,3
424.175,SUSR,N,$D_FLARE,3
424.214,SUSR,N,$D_TGT,30
424.254,SUSR,N,$D_ABORT,1090
424.296,SUSR,N,$D_NO_BLEED,200
424.338,SUSR,N,$D_BOOST,0
424.374,SUSR,N,$T_BOOST,0
424.409,SUSR,N,$D_FINISH,0
424.446,SUSR,N,$D_PITCH,0
424.481,SUSR,N,$D_SAFE,0
424.516,SUSR,N,$D_CALL,0
424.550,SUSR,N,$SURFACE_URGENCY,0
424.595,SUSR,N,$SURFACE_URGENCY_TRY,0
424.645,SUSR,N,$SURFACE_URGENCY_FORCE,0
```

```
424.700,SUSR,N,$T_DIVE,10
424.740,SUSR,N,$T_MISSION,15
424.785,SUSR,N,$T_ABORT,1440
424.828,SUSR,N,$T_TURN,225
424.868,SUSR,N,$T_TURN_SAMPINT,5
424.917,SUSR,N,$T_NO_W,120
424.954,SUSR,N,$T_LOITER,0
424.994,SUSR,N,$USE_BATHY,-4
425.033,SUSR,N,$USE_ICE,0
425.074,SUSR,N,$ICE_FREEZE_MARGIN,0.30000001
425.135,SUSR,N,$D_OFFGRID,100
425.179,SUSR,N,$T_WATCHDOG,10
425.222,SUSR,N,$RELAUNCH,1
425.262,SUSR,N,$APOGEE_PITCH,-5
425.308,SUSR,N,$MAX_BUOY,150
425.347,SUSR,N,$COURSE_BIAS,0
425.391,SUSR,N,$GLIDE_SLOPE,30
425.435,SUSR,N,$SPEED_FACTOR,1
425.481,SUSR,N,$RHO,1.026
425.522,SUSR,N,$MASS,53188
425.562,SUSR,N,$NAV_MODE,1
425.602,SUSR,N,$FERRY_MAX,45
425.641,SUSR,N,$KALMAN_USE,0
425.686,SUSR,N,$HD_A,0.003
425.729,SUSR,N,$HD_B,0.0099999998
425.780,SUSR,N,$HD_C,9.9999997e-06
425.829,SUSR,N,$HEADING,-1
425.866,SUSR,N,$ESCAPE_HEADING,0
425.913,SUSR,N,$ESCAPE_HEADING_DELTA,10
426.291,SUSR,N,$FIX_MISSING_TIMEOUT,0
426.346,SUSR,N,$TGT_DEFAULT_LAT,4736
426.400,SUSR,N,$TGT_DEFAULT_LON,-12218
426.451,SUSR,N,$TGT_AUTO_DEFAULT,0
426.501,SUSR,N,$SM_CC,500
426.539,SUSR,N,$N_FILEKB,4
426.576,SUSR,N,$FILEMGR,0
426.614,SUSR,N,$CALL_NDIVES,1
426.655,SUSR,N,$COMM_SEQ,0
426.692,SUSR,N,$KERMIT,0
426.729,SUSR,N,$N_NOCOMM,1
426.765,SUSR,N,$N_NOSURFACE,0
426.808,SUSR,N,$UPLOAD_DIVES_MAX,-1
426.859,SUSR,N,$CALL_TRIES,5
426.901,SUSR,N,$CALL_WAIT,60
426.943,SUSR,N,$CAPUPLOAD,1
426.984,SUSR,N,$CAPMAXSIZE,100000
```

```
427.029,SUSR,N,$HEAPDBG,0
427.069,SUSR,N,$T_GPS,15
427.107,SUSR,N,$N_GPS,20
427.141,SUSR,N,$T_GPS_ALMANAC,0
427.189,SUSR,N,$T_GPS_CHARGE,-143.86217
427.244,SUSR,N,$T_RSLEEP,3
427.281,SUSR,N,$STROBE,0
427.318,SUSR,N,$RAFOS_PEAK_OFFSET,1.5
427.371,SUSR,N,$RAFOS_CORR_THRESH,60
427.425,SUSR,N,$RAFOS_HIT_WINDOW,3600
427.480,SUSR,N,$PITCH_MIN,115
427.526,SUSR,N,$PITCH_MAX,3879
427.573,SUSR,N,$C_PITCH,1591
427.618,SUSR,N,$PITCH_DBAND,0.1
427.666,SUSR,N,$PITCH_CNV,0.003125763
427.722,SUSR,N,$P_OVSHOOT,0.039999999
427.775,SUSR,N,$PITCH_GAIN,30
427.821,SUSR,N,$PITCH_TIMEOUT,16
427.868,SUSR,N,$PITCH_AD_RATE,175
427.916,SUSR,N,$PITCH_MAXERRORS,1
427.961,SUSR,N,$PITCH_ADJ_GAIN,0
428.005,SUSR,N,$PITCH_ADJ_DBAND,0
428.055,SUSR,N,$ROLL_MIN,231
428.100,SUSR,N,$ROLL_MAX,3792
428.144,SUSR,N,$ROLL_DEG,40
428.187,SUSR,N,$C_ROLL_DIVE,2804
428.237,SUSR,N,$C_ROLL_CLIMB,2804
428.285,SUSR,N,$HEAD_ERRBAND,10
428.333,SUSR,N,$ROLL_CNV,0.028270001
428.386,SUSR,N,$ROLL_TIMEOUT,15
428.434,SUSR,N,$R_PORT_OVSHOOT,22
428.482,SUSR,N,$R_STBD_OVSHOOT,19
428.531,SUSR,N,$ROLL_AD_RATE,350
428.577,SUSR,N,$ROLL_MAXERRORS,1
428.621,SUSR,N,$ROLL_ADJ_GAIN,0
428.664,SUSR,N,$ROLL_ADJ_DBAND,0
428.713,SUSR,N,$VBD_MIN,491
428.757,SUSR,N,$VBD_MAX,3961
428.801,SUSR,N,$C_VBD,2966
428.841,SUSR,N,$VBD_DBAND,2
428.885,SUSR,N,$VBD_CNV,-0.245296
428.935,SUSR,N,$VBD_TIMEOUT,720
428.983,SUSR,N,$PITCH_VBD_SHIFT,0.0012300001
429.044,SUSR,N,$VBD_PUMP_AD_RATE_SURFACE,5
429.103,SUSR,N,$VBD_PUMP_AD_RATE_APOGEE,4
```

```
429.160,SUSR,N,$VBD_BLEED_AD_RATE,8
429.211,SUSR,N,$UNCOM_BLEED,20
429.582,SUSR,N,$VBD_MAXERRORS,1
429.628,SUSR,N,$CF8_MAXERRORS,20
429.676,SUSR,N,$AH0_24V,150
429.717,SUSR,N,$AH0_10V,100
429.759,SUSR,N,$MINV_24V,19
429.800,SUSR,N,$MINV_10V,8
429.837,SUSR,N,$FG_AHR_10V,0
429.876,SUSR,N,$FG_AHR_24V,0
429.918,SUSR,N,$PHONE_SUPPLY,2
429.965,SUSR,N,$PRESSURE_YINT,-53.006554
430.024,SUSR,N,$PRESSURE_SLOPE,0.000011597844
430.087,SUSR,N,$AD7714Ch0Gain,128
430.132,SUSR,N,$TCM_PITCH_OFFSET,0
430.178,SUSR,N,$TCM_ROLL_OFFSET,0
430.223,SUSR,N,$COMPASS_USE,0
430.264,SUSR,N,$ALTIM_BOTTOM_PING_RANGE,0
430.318,SUSR,N,$ALTIM_TOP_PING_RANGE,0
430.374,SUSR,N,$ALTIM_BOTTOM_TURN_MARGIN,12
430.431,SUSR,N,$ALTIM_TOP_TURN_MARGIN,0
430.486,SUSR,N,$ALTIM_TOP_MIN_OBSTACLE,1
430.542,SUSR,N,$ALTIM_PING_DEPTH,80
430.593,SUSR,N,$ALTIM_PING_DELTA,5
430.644,SUSR,N,$ALTIM_FREQUENCY,13
430.693,SUSR,N,$ALTIM_PULSE,3
430.736,SUSR,N,$ALTIM_SENSITIVITY,2
430.787,SUSR,N,$XPDR_VALID,2
430.829,SUSR,N,$XPDR_INHIBIT,90
430.877,SUSR,N,$INT_PRESSURE_SLOPE,0.0097660003
430.944,SUSR,N,$INT_PRESSURE_YINT,-0.28
430.996,SUSR,N,$DEEPGLIDER,0
431.035,SUSR,N,$DEEPGLIDERMB,0
431.080,SUSR,N,$MOTHERBOARD,4
431.125,SUSR,N,$DEVICE1,83
431.165,SUSR,N,$DEVICE2,-1
431.205,SUSR,N,$DEVICE3,-1
431.244,SUSR,N,$DEVICE4,-1
431.284,SUSR,N,$DEVICE5,-1
431.323,SUSR,N,$DEVICE6,-1
431.363,SUSR,N,$LOGGERS,1
431.402,SUSR,N,$LOGGERDEVICE1,85
431.448,SUSR,N,$LOGGERDEVICE2,-1
431.495,SUSR,N,$LOGGERDEVICE3,-1
431.541,SUSR,N,$LOGGERDEVICE4,-1
431.591,SUSR,N,$COMPASS_DEVICE,33
431.639,SUSR,N,$COMPASS2_DEVICE,-1
```

```
431.690,SUSR,N,$PHONE_DEVICE,48
431.738,SUSR,N,$GPS_DEVICE,32
431.781,SUSR,N,$RAFOS_DEVICE,-1
431.829,SUSR,N,$XPDR_DEVICE,24
431.871,SUSR,N,$SIM_W,0
431.907,SUSR,N,$SIM_PITCH,-20
431.953,SUSR,N,$SEABIRD_T_G,0.0040000002
432.011,SUSR,N,$SEABIRD_T_H,0.000600000003
432.071,SUSR,N,$SEABIRD_T_I,2.4000001e-05
432.131,SUSR,N,$SEABIRD_T_J,2.6e-06
432.184,SUSR,N,$SEABIRD_C_G,-11
432.232,SUSR,N,$SEABIRD_C_H,0.60000002
432.288,SUSR,N,$SEABIRD_C_I,-0.0014
432.342,SUSR,N,$SEABIRD_C_J,9.9999997e-05
432.400,SSENSOR,N,$PC_RECORDABOVE,1000.0
432.455,SSENSOR,N,$PC_PROFILE,3.0
432.881,SSENSOR,N,$PC_XMITPROFILE,3.0
432.935,SSENSOR,N,$PC_UPLOADMAX,10000000.0
432.991,SSENSOR,N,$PC_STARTS,4.0
433.037,SSENSOR,N,$PC_INTERVAL,1.0
434.770,SSURF,N,Sending cmd lsx -k -t200 targets
444.852,SSURF,N,No targets file on basestation [lsx -k -t200 targets
lsx: cannot open targets: No such file or directory

Can't open any requested files.
sg535= ]
448.810,SSURF,N,Sending cmd lsx -k -t200 science
450.666,SSURF,N,Receiving science...
452.128,SSURF,N,SOH/STX: 0x1 received (3), secSize= 128...
454.196,SSURF,N,EOT: 0x4 received (1), secSize= 128...

454.862,SSURF,N,Received science 115 bytes
455.818,SSURF,N,Transmission succeeded...
456.635,SSURF,N,Sending cmd lsx -k -t200 pdoscmds.bat
466.723,SSURF,N,No pdoscmds.bat file on basestation [echo pdoscmds.bat >> comm.log
sg535= lsx -k -t200 pdoscmds.bat
lsx: cannot open pdoscmds.bat: No such file or directory
Can't open any requested files.
sg535= ]
467.187,SSURF,N,Sending cmd lrx -y -c -t200 st0002lu.x00
468.984,SSURF,N,Sending st0002lu.x00...
469.085,SSURF,N,block: 1, size: 1024, blkSize: 1024, attempt: 1...
475.553,SSURF,N,block: 2, size: 1024, blkSize: 1024, attempt: 1...
481.852,SSURF,N,block: 3, size: 1024, blkSize: 1024, attempt: 1...
488.062,SSURF,N,block: 4, size: 1024, blkSize: 1024, attempt: 1...
494.271,SSURF,N,EOT sent ...
```

```
496.791,SSURF,N,recieved ACK (attempt 0)
500.213,SSURF,N,Transmission succeeded...
500.377,SSURF,N,Sending cmd lrx -y -c -t200 st0002lu.x01
502.310,SSURF,N,Sending st0002lu.x01...
502.392,SSURF,N,block: 1, size: 174, blkSize: 1024, attempt: 1...
508.670,SSURF,N,EOT sent ...
511.192,SSURF,N,recieved ACK (attempt 0)
514.614,SSURF,N,Transmission succeeded...
514.719,SSURF,N,Sending cmd lrx -y -c -t200 st0002du.x00
516.528,SSURF,N,Sending st0002du.x00...
516.614,SSURF,N,block: 1, size: 930, blkSize: 1024, attempt: 1...
522.799,SSURF,N,EOT sent ...
525.230,SSURF,N,recieved ACK (attempt 0)
526.853,SSURF,N,Transmission succeeded...
526.989,SSURF,N,Sending cmd lrx -y -c -t200 pc0003az.x00
529.129,SSURF,N,Sending pc0003az.x00...
529.211,SSURF,N,block: 1, size: 371, blkSize: 1024, attempt: 1...
535.489,SSURF,N,EOT sent ...
538.010,SSURF,N,recieved ACK (attempt 0)
541.523,SSURF,N,Transmission succeeded...
541.626,SSURF,N,Sending cmd lrx -y -c -t200 pc0003bz.x00
543.528,SSURF,N,Sending pc0003bz.x00...
543.610,SSURF,N,block: 1, size: 364, blkSize: 1024, attempt: 1...
549.800,SSURF,N,EOT sent ...
552.232,SSURF,N,recieved ACK (attempt 0)
553.853,SSURF,N,Transmission succeeded...
Compressing THISDIVE.KAP to st0002kz.x...
558.018,SSURF,N,Sending cmd lrx -y -c -t200 st0002kz.x
559.816,SSURF,N,Sending st0002kz.x...
559.962,SSURF,N,block: 1, size: 1024, blkSize: 1024, attempt: 1...
566.270,SSURF,N,block: 2, size: 1024, blkSize: 1024, attempt: 1...
572.480,SSURF,N,block: 3, size: 1024, blkSize: 1024, attempt: 1...
578.781,SSURF,N,block: 4, size: 1024, blkSize: 1024, attempt: 1...
585.080,SSURF,N,block: 5, size: 1024, blkSize: 1024, attempt: 1...
591.379,SSURF,N,block: 6, size: 1024, blkSize: 1024, attempt: 1...
597.680,SSURF,N,block: 7, size: 1024, blkSize: 1024, attempt: 1...
603.982,SSURF,N,block: 8, size: 1024, blkSize: 1024, attempt: 1...
610.280,SSURF,N,block: 9, size: 1024, blkSize: 1024, attempt: 1...
616.579,SSURF,N,block: 10, size: 1024, blkSize: 1024, attempt: 1...
622.971,SSURF,N,block: 11, size: 1024, blkSize: 1024, attempt: 1...
629.272,SSURF,N,block: 12, size: 864, blkSize: 1024, attempt: 1...
635.570,SSURF,N,EOT sent ...
638.091,SSURF,N,recieved ACK (attempt 0)
641.511,SSURF,N,Transmission succeeded...
```

```
643.943,SSURF,N,logout...
654.007,SPOWER,N,powerOFF (7,Iridium_during_xfer) without corresponding powerON!
654.111,SSURF,N,Total NAK's: 0, timeouts: 0
654.265,SGLMALLOC,N,glheap_walk: 322824 bytes free, 12 blocks free, 34160 bytes alloc, 47 blocks alloc
654.444,SUSR,N,---- Self test FAILED or ABORTED! ----
654.552,SUSR,N,1 failures noted
654.630,SUSR,N,--> bathymetry maps failed
654.726,SUSR,N,Restoring original settings...
```

APPENDIX F

# *Hardware and Configuration Menus*

There are a number of functional checks of, and changes to, configuration that can be done when directly connected to the 1KA Seaglider via the communication cable. All actions are found under the Main Menu. The content of each sub-menu of the Main Menu are shown in the figures below along with a brief description.

**Note:** These menus can also be accessed through the pdoscmds.bat file which is uploaded by Seaglider when it calls into the basestation. For information on how to use the pdoscmds.bat file method see the "Extended PicoDOS® Reference Manual, v66.07" on page 303.

To access one of the sub-menus of the Main Menu, Figure F-1, enter the number found in front of the parameter name and press ENTER.

**FIGURE F-1. Main Menu**

```
------ Main Menu ------
     1 [param   ]  Parameters and configuration
     2 [hw      ]  Hardware tests and monitoring
     3 [modes   ]  Test operation modes and files
     4 [pdos    ]  PicoDOS commands (and exit)
     5 [launch  ]  Pre-launch
|
```

The main menu is characteristic of all menus on the glider. Selections can either be made numerically or by typing the name of the shortcut contained inside the [ ].

The advantage to using names versus numbers is that they do not typically change when the menu items are rearranged or items are added when the software version changes.

Using names also facilitates navigating multiple menus with a single command. Sub-menus can be accessed by adding a / and the name of the sub-menu. For example from the toplevel menu it is possible to do a compass selftest by typing the following text at the main menu prompt:

Enter selection (1-5,CR): **hw/compass/selftest**

Navigating one menu at a time requires typing **hw** (or the number 2) which then displays the hardware test menu. At the hardware menu prompt you would type **compass** (or the number **6**) to display the compass menu. Finally you would type **selftest** (or the number **1**) to run the actual selftest. This works at any level of the menu tree. If you are already at the hardware menu, typing **compass/selftest** would run the selftest item from the compass sub-menu.

The entire menu structure can be displayed by typing **help** at the main menu prompt. Typing **help** at any prompt will display the menu structure below that point in the menu tree. Typing **help search** will display all menu entries below that point in the tree containing the word search in their names or labels.

Many menu system functions require additional user input. These inputs can generally be provided as either arguments to the selection command entered at the menu prompt or as answers to separate questions asked once the function is selected. For example, if you select the pressure sensor self test as shown below, you will be prompted to answer yes or no as to whether the glider is at sea level.

Enter selection (1-5,CR): **hw/pressure/selftest**

Alternatively, you could provide this information at the menu prompt as follows:

Enter selection (1-5,CR): **hw/pressure/selftest sealevel=1**

In this case sea level=1 indicates that the glider is at sea level. A value of 0 would indicate the negative response.

Using optional arguments to the menu selection prompt is the only way to provide information to menu functions when using the menu system non-interactively via

extended PicoDOS. Functions in the menu tree can be accessed via extended PicoDOS using the "menu" command as shown in this example:

picoDOS>> **menu hw/pressure/selftest sea level=1**

Using the pdoscmds.bat provides a powerful mechanism for accessing test functionality while the glider is operating in the field.

In addition to menu selections you can also execute extended PicoDOS commands and parameter changes at any menu prompt. PicoDOS commands are preceded by a "!". For example, tying the command below, displays the directory of the compact flash card:

Enter selection (1-5,CR): **!dir**

You can also reach the extended PicoDOS prompt (picoDOS>>) from any menu by typing **pdos**. Parameters can be viewed or changed simply by typing their full name with leading "$":

This command would display the value of $T_DIVE:

Enter selection (1-5,CR): **$T_DIVE**

While this command would change the value of $T_DIVE to 30.

Enter selection (1-5,CR): **$T_DIVE,30**

## *Parameters and Configuration Menu*

To view the Parameters and Configuration menu, Figure F-2:

- Select **1** and press ENTER

**FIGURE F-2. Parameters and Configuration**

```
------ Edit parameters ------
*Flight control and mission definition
     1 [basic   ]  Basic mission and glider parameters
     2 [dive    ]  Dive parameters
     3 [flight  ]  Flight parameters
     4 [surface ]  Surface parameters
     5 [rafos   ]  RAFOS parameters
     6 [password]  Set/show glider login password
     7 [telnum  ]  Set/show basestation phone number
     8 [altnum  ]  Set/show basestation alternate phone number
*Pitch, roll, VBD
     9 [pitch   ]  Pitch parameters
    10 [roll    ]  Roll parameters
    11 [vbd     ]  VBD parameters
*Sensors and peripherals
    12 [config  ]  Hardware configuration parameters
    13 [pressure]  Pressure (external) parameters
    14 [intpress]  Pressure (internal) parameters
    15 [compass ]  Compass paramaters
    16 [altim   ]  Altimetry parameters
    17 [seabird ]  Seabird CT calibration
    18 [power   ]  Power parameters
*Utility
    19 [all     ]  Edit all parameters
    20 [validate]  Validate parameters
    21 [details ]  Show parameter details
    22 [show    ]  Show changed parameters
    23 [clear   ]  Clear changed parameters
    24 [save    ]  Save parameters by name to file
    25 [dump    ]  Dump parameters to screen
    26 [load    ]  Load parameters from file
    27 [reset   ]  Reset to defaults
    CR)  Return to previous
```

## *Basic Mission and Seaglider Parameters*

To access the Basic Mission and Seaglider Parameters menu:

- Select **1**: Basic Mission and Seaglider Parameters and press ENTER

  The menu in Figure F-3 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

  The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83.

**FIGURE F-3. Basic Mission and Seaglider Parameters Menu**

```
Enter selection (1-27,CR): 1
------ Basic (current value in [])-------
     1) $ID [506] Glider ID (3 digits max)
     2) $MISSION [2] Current mission number
     3) $T_WATCHDOG [10] Watchdog timer (min)
     4) $RELAUNCH [0] Relaunch glider if crash to TOM8?
        (0=no/1=yes)
     5) $HEAPDBG [0] In depth heap debugging  (0=off/1=on)
     6) $CF8_MAXERRORS [20] Number of CF8 errors allowed before
        entering recovery
    CR) Return to previous menu
|
```

To change the value for a parameter:

- Enter the number found in front of the parameter name and press ENTER.
- When prompted, type in the desired value and press ENTER.

Example:

If the mission number needs to be edited from 2 to 3, select **2**: $MISSION and press ENTER

Seaglider responds with:

New value? [2.00]  Press ENTER if correct. The user is returned to the Basic Mission and Seaglider Parameters menu.

If the value is incorrect, when Seaglider responds with:

New value? [2.00]  type 3 and press ENTER (Figure F-4).

Seaglider acknowledges the update and the user is returned to the Basic Mission and Seaglider Parameters menu.

**FIGURE F-4. Checking/Changing Seaglider's Mission Number**

```
$MISSION,
New value? [2.00] 3
1373.601,HTT8,N,Updating parameter $MISSION to 3
```

- To exit the Basic menu and return to the Parameters and Configuration menu, press ENTER.

## *Dive Parameters*

To access the Dive Parameters menu from the Parameters and Configuration menu:

- Select **2**: Dive Parameters and press ENTER

  The menu in Figure F-5 is displayed. To edit a parameter value, follow the procedures under "Basic Mission and Seaglider Parameters" on page 371.

**FIGURE F-5. Dive Parameters Menu**

```
------ Dive (current value in [])-------
     1) $DIVE [9] Next dive number
     2) $D_FLARE [3] Flare depth (m)
     3) $D_TGT [10] Target depth (m)
     4) $D_ABORT [1090] Abort depth (m)
     5) $D_NO_BLEED [500] Depth below which we will not bleed on dives (m)
     6) $D_BOOST [0] Depth above which only boost pump is run (m)
     7) $T_BOOST [0] Time to run the boost pump before turning off (0 do not turn off)
     8) $T_DIVE [15] Target time per dive/climb cycle (min)
     9) $T_MISSION [15] Maximum mission time per dive/climb cycle (min)
    10) $T_ABORT [1440] Failsafe watchdog  mission time - change with extreme care (min)
    11) $T_TURN [225] Max time allowed for a turn maneuver (sec)
    12) $T_TURN_SAMPINT [4] Sample interval during a turn (int sec, >=4)
    13) $T_NO_W [60] Time with no vertical velocity (w) before taking next step (secs)
    14) $T_LOITER [0] Time to loiter at apogee before climbing (secs)
    15) $USE_BATHY [-1] Use bathymetry map (0 - no, +/-nnn - file number)
    16) $D_OFFGRID [100] Max target depth if outside bathymap (m)
    17) $SIM_W [0.1] Simulated W (m/s) 0 - use pressure gauge to compute depth
    18) $SIM_PITCH [-20] Simulated dive pitch (e.g., -20 degrees) 0 - use TCM2 to report pitch
    CR) Return to previous menu
Enter selection (1-18,CR):
```

## *Flight Parameters*

To access the Flight Parameters menu from the Parameters and Configuration menu:

- Select **3**: Flight Parameters and press ENTER

  The menu in Figure F-6 is displayed. To edit a parameter value, follow the procedures under “Basic Mission and Seaglider Parameters” on page 371.

**FIGURE F-6. Flight Parameters Menu**

```
------ Flight (current value in [])-------
     1) $D_SAFE [0] Safety dive depth for emergency/escape maneuvers
        (m)
     2) $APOGEE_PITCH [-8] Intermediate pitch angle @ apogee (deg)
     3) $MAX_BUOY [220] Max buoyancy force allowed @ apogee (g)
     4) $COURSE_BIAS [0] Course bias to subtract from the computed
        (deg, +cw)
     5) $GLIDE_SLOPE [30] Max glide slope allowed (deg)
     6) $SPEED_FACTOR [1] Fudge factor for glider speed
     7) $RHO [1.0233001] Water density (grams/cc)
     8) $MASS [51509] Glider mass (g)
     9) $NAV_MODE [1] Navigation mode (0=direct/1=Kalman/2=Set
        correct/3=Current relative)
    10) $FERRY_MAX [45] Ferry angle bound for set correction navigation
        (deg)
    11) $KALMAN_USE [2] Kalman filter (0=reset/1=enable/2=skip)
    12) $HD_A [0.003] Glider hydrodynamic lift (A)
    13) $HD_B [0.0099999998] Glider hydrodynamic drag (B)
    14) $HD_C [9.9999997e-06] Glider hydrodynamic induced drag (C)
    15) $HEADING [-1] Heading (degrees true) to steer (-1 disable)
    16) $ESCAPE_HEADING [0] Escape heading (degrees magnetic) to steer
        in emergency
    17) $ESCAPE_HEADING_DELTA [10] Escape heading offset (degrees) in
        emergency
    18) $FIX_MISSING_TIMEOUT [0] Time (days) without fix before
        activating escape heading
    19) $TGT_DEFAULT_LAT [4808] Default target latitude
    20) $TGT_DEFAULT_LON [-12223] Default target longitude
    21) $TGT_AUTO_DEFAULT [0] Automatically update default target
    CR) Return to previous menu
Enter selection (1-21,CR):
```

## *Surface Parameters*

To access the Surface Parameters menu from the Parameters and Configuration menu:

- Select **4**: Surface Parameters and press ENTER

  The menu in Figure F-7 is displayed. To edit a parameter value, follow the procedures under "Basic Mission and Seaglider Parameters" on page 371.

**FIGURE F-7. Surface Parameters Menu**

```
------ Parameters (current value in [])-------
     1) $D_SURF [4] Surface depth (m)
     2) $D_FINISH [10] Dive finish depth (m)
     3) $D_PITCH [0] Depth to achieve before surface maneuver pitch (m)
     4) $D_CALL [0] Depth to achieve before antenna is assumed to be
        out of water (m)
     5) $SURFACE_URGENCY [0] Number of dives to accumulate before
        trying extra surfacings
     6) $SURFACE_URGENCY_TRY [0] Dive number modulo for extra surfacing
        attempts
     7) $SURFACE_URGENCY_FORCE [0] Dive number modulo for forced extra
        surfacing attempts
     8) $USE_ICE [0] Use ice map for surfacing decisions (0 - no, +/-
        nnn - file number)
     9) $ICE_FREEZE_MARGIN [0.30000001] Margin (degC) in freezing point
        check for surfacing
    10) $SM_CC [520] VBD volume for surfacing (cc)
    11) $N_FILEKB [4] File segment size for upload (KBytes <=16, 0=no
        split, negative=no gzip)
    12) $FILEMGR [0] File management aggressiveness: 0=none, 1=only
        store compressed, 2=delete splits on failed phone call
    13) $CALL_NDIVES [1] # of dive/climb cycles before calling
    14) $COMM_SEQ [0] Comm sequence to use (0-default, 1-ctl only, 2-no
        loggers)
    15) $KERMIT [0] Use of kermit (0 - none, 1 - data only, 2 - data
        only, batch, 3 data and control, 4 data and control - data
        batch)
    16) $N_NOCOMM [1] # dives without communications before max surface
    17) $N_NOSURFACE [0] modulo for dives to finish under surface
    18) $UPLOAD_DIVES_MAX [-1] maximum # of dives to upload at each
        surfacing (-1 = all available)
    19) $CALL_TRIES [5] # of calls to try at each surfacing
    20) $CALL_WAIT [60] Delay between phone tries (sec)
    21) $CAPUPLOAD [1] Upload capture file (0=no/1=yes)
    22) $CAPMAXSIZE [200000] Maximum capture file size to upload
        (before compression)
    23) $T_GPS [15] GPS timeout (min)
    24) $N_GPS [20] Max # valid GPS readings
    25) $T_GPS_ALMANAC [0] Time to wait to reacquire the GPS almanac
        (mins)
    26) $T_GPS_CHARGE [-10258.504] How long to trickle charge the GPS
        (secs)
    27) $T_RSLEEP [3] Sleep interval during recovery (min)
    28) $STROBE [0] Strobe control (0=never, 1=recovery, 2=surface)
    CR) Return to previous menu
    Enter selection (1-28,CR
```

## *RAFOS Parameters*

To access the RAFOS Parameters menu from the Parameters and Configuration menu:

- Select **5**: RAFOS Parameters and press ENTER

  The menu in Figure F-8 is displayed.

**Note:** RAFOS is not licensed to iRobot and as such, while these parameters do appear in parameter lists, they are NOT activated in the iRobot version of 1KA Seaglider software.

**FIGURE F-8. RAFOS Parameters Menu**

```
------ Parameters (current value in [])-------
        1) $RAFOS_PEAK_OFFSET [1.5] internal delay of RAFOS receiver
           (seconds)
        2) $RAFOS_CORR_THRESH [60] threshold correlation value for hits
           used in navigation
        3) $RAFOS_HIT_WINDOW [3600] window length (seconds) in which to
           search for hits to use in navigation
   <CR) Return to previous menu
Enter selection (1-3,CR):
```

- To exit the RAFOS Parameters menu and return to the Parameters and Configuration menu, press ENTER.

## *Password*

To access Seaglider's password from the Parameters and Configuration menu:

1. Select **6**: Password and press ENTER

   The present password is displayed. Figure F-9.

   When prompted for a new password, verify the value of the present password. The password assigned to a Seaglider at the factory is 6 digits long. The first three digits are Seaglider's ID number. The last three digits are either 680 if Seaglider's ID is even or 791 if Seaglider's ID is odd. However, passwords can be any alphanumeric string, up to 15 characters long. Passwords cannot contain punctuation or special characters. The password in Seaglider must be the same as what is on the basestation for this Seaglider.

2. One of the following occurs:
   - If the password is correct press ENTER. The user is returned to the Parameters and Configuration menu.
   - If the password is incorrect enter the correct password and press ENTER.

   Seaglider acknowledges the update and the user is returned to the Parameters and Configuration menu.

   If there are any questions contact iRobot Customer Support. A Seaglider cannot call in to the basestation unless the password is the same on both the basestation and Seaglider.

**FIGURE F-9. Seaglider Password Check**

```
6477.808,SUSR,N,Current password is 506680
New password (15 char max length, CR to leave unchanged): |
```

## *Telnum*

To access the Basestation Telephone Number from the Parameters and Configuration menu:

1. Select **7**: Telnum and press ENTER

   The present basestation telephone number is displayed. Figure F-10.

Do one of the following:

- If the telephone number is correct press ENTER. The user is returned to the Parameters and Configuration menu.
- If the telephone number is incorrect type the correct number and press ENTER. Telephone numbers can be no more than 15 digits long. For more information see Chapter 3, “Setting Up the System” on page 33.

Seaglider acknowledges the update and the user is returned to the Parameters and Configuration menu.

**FIGURE F-10. Verifying Basestation Telephone Number in Seaglider**

```
6561.080,SUSR,N,Current telnum is 12062215341
New telnum (15 char max length, CR to leave unchanged):
    19194841429
6605.038,SUSR,N,Changing telnum to 19194841429|
```

## *Altnum*

To access the Alternate Basestation Telephone Number from the Parameters and Configuration menu:

- Select **8**: Altnum and press ENTER

The present alternate basestation telephone number is displayed. Figure F-11.

**FIGURE F-11. Verifying Basestation Alternate Telephone Number in Seaglider**

```
6699.777,SUSR,N,Current altnum is 12062217301
New altnum (15 char max length, CR to leave unchanged):
19193612847
6725.312,SUSR,N,Changing altnum to 19193612847
```

To edit a parameter value, follow the procedures in “Telnum” on page 376.

## *Pitch Parameters*

To access the Pitch Parameters menu from the Parameters and Configuration menu:

- Select **9**: Pitch Parameters and press ENTER

The menu in Figure F-12 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $PITCH_MIN, $PITCH_MAX and $C_PITCH that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

**FIGURE F-12. Pitch Parameters Menu**

```
------ Pitch (current value in [])-------
        1) $PITCH_MIN [160 (-9.4 cm)] Pitch position limit, fwd (AD#)
        2) $PITCH_MAX [3940 (2.4 cm)] Pitch position limit, aft (AD#)
        3) $C_PITCH [3180] Pitch neutral point, (AD#)
        4) $PITCH_DBAND [+/- 0.10] Pitch position deadband (cm)
        5) $PITCH_CNV [0.003125763] Pitch position conversion factor
           (cm/AD#)
        6) $P_OVSHOOT [0.039999999] Pitch mass overshoot after motor off
           (cm)
        7) $PITCH_GAIN [26] Pitch angle to mass gain (deg/cm)
        8) $PITCH_TIMEOUT [17] Pitch mass timeout (sec)
        9) $PITCH_AD_RATE [175] Pitch motor AD counts per sec
        10)$PITCH_MAXERRORS [1] Number of pitch errors allowed before
           entering recovery
        11) $PITCH_ADJ_GAIN [0] Gain for auto-adjusting pitch (cm/deg)
        12) $PITCH_ADJ_DBAND [0] Deadband for auto-adjusting pitch (deg)
        CR) Return to previous menu
   Enter selection (1-12,CR):
```

To change a parameter value, follow the procedures under “Basic Mission and Seaglider Parameters” on page 371.

## *Roll Parameters*

To access the Roll Parameters menu from the Parameters and Configuration menu:

- Select **10**: Roll Parameters and press ENTER

The menu in Figure F-13 is displayed.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $ROLL_MIN, $ROLL_MAX and $C_ROLL that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

To change a parameter value, follow the procedures “Basic Mission and Seaglider Parameters” on page 371.

**FIGURE F-13. Roll Parameters**

```
------ Roll (current value in [])-------
  1) $ROLL_MIN [180 (-59.6 deg)] Roll position limit, port (AD#)
  2) $ROLL_MAX [3875 (44.8 deg)] Roll position limit, stbd (AD#)
  3) $ROLL_DEG [45] Roll position normal flight (deg)
  4) $C_ROLL_DIVE [2290] Roll neutral point when diving and testing,
     (AD#)
  5) $C_ROLL_CLIMB [2090] Roll neutral point when climbing, (AD#)
  6) $HEAD_ERRBAND [+/- 10.00] Roll (scene)/Heading (auto)
     deadband,(deg)
  7) $ROLL_CNV [0.028270001] Roll position conversion factor
     (deg/AD#)
  8) $ROLL_TIMEOUT [15] Roll mass timeout, (roll and rollback, sec)
  9) $R_PORT_OVSHOOT [18] Roll mass overshoot to port after motor
     off (AD)
 10) $R_STBD_OVSHOOT [28] Roll mass overshoot to stbd after motor
     off (AD)
 11) $ROLL_AD_RATE [400] Roll motor AD count per sec
 12) $ROLL_MAXERRORS [1] Number of roll errors allowed before
     entering recovery
 13) $ROLL_ADJ_GAIN [1.5] Gain for auto-adjusting roll center (deg
     battery roll / deg/sec turn rate
 14) $ROLL_ADJ_DBAND [0.025] Deadband for auto-adjusting roll center
     (deg/sec)
 CR) Return to previous menu |
```

## *VBD Parameters*

To access the VBD Parameters menu from the Parameters and Configuration menu:

- Select **11**: VBD Parameters and press ENTER

The menu in Figure F-14 is displayed.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $VBD_MIN, $VBD_MAX and $C_VBD that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

To change a parameter value, follow the procedures in “Basic Mission and Seaglider Parameters” on page 371.

**FIGURE F-14. VBD Parameters Menu**

```
------ VBD (current value in [])-------
        1) $VBD_MIN [430 (654.9 cc)] VBD bladder position limit, minimum
           (AD$)
        2) $VBD_MAX [3959 (-210.7 cc)] VBD bladder position limit,
           maximum (AD$)
        3) $C_VBD [3100] VBD bladder neutral point, (AD$)
        4) $VBD_DBAND [+/- 2.00] VBD bladder position deadband (cc)
        5) $VBD_CNV [-0.245296] VBD bladder position conversion factor
           (cc/AD$)
        6) $VBD_TIMEOUT [720] VBD timeout (sec)
        7) $PITCH_VBD_SHIFT [0.0012300001] Pitch offset due to VBD
           (cm/cc)
        8) $VBD_PUMP_AD_RATE_SURFACE [5] VBD pump AD counts per sec at
           surface
        9) $VBD_PUMP_AD_RATE_APOGEE [4] VBD pump AD counts per sec at
           apogee
       10) $VBD_BLEED_AD_RATE [8] VBD bleed AD counts per sec at surface
       11) $UNCOM_BLEED [50] Uncommanded bleed change (AD) 0 - disabled
       12) $VBD_MAXERRORS [1] Number of VBD errors allowed before
           entering recovery
     ?R) Return to previous menu
```

## *Hardware Configuration Parameters*

To access the Hardware Configuration Parameters menu from the Parameters and Configuration menu:

- Select **12**: Hardware Configuration Parameters and press ENTER

The menu in Figure F-15 is displayed. Seaglider hardware configuration is done at the factory. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

**FIGURE F-15. Hardware Configuration Parameters Menu**

```
------ Edit hardware configuration parameters ------
     1 [show    ]  Show current hardware configuration
     2 [motherbo]  Configure motherboard
     3 [compass ]  Configure compass
     4 [compass2]  Configure spare compass
     5 [phone   ]  Configure phone
     6 [gps     ]  Configure GPS
     7 [xpdr    ]  Configure XPDR
     8 [sensor  ]  Configure sensor
     9 [logger  ]  Configure logger sensor
    10 [add     ]  Add device file to hardware library
    11 [logdev  ]  Add logger file to hardware library
    12 [pressure]  Configure external pressure sensor
    13 [param   ]  Edit parameters directly
    CR)  Return to previous
```

- To exit the Hardware Configuration Parameters menu and return to the Parameters and Configuration menu, press ENTER.

Seaglider responds with:

```
Re-initialize hardware configuration [N]
```

Type [N] unless hardware configuration changes have been made with the blessing of iRobot.

## *Pressure (external) Parameters*

To access the External Pressure Parameters menu from the Parameters and Configuration menu:

- Select **13**: Pressure (external) Parameters and press ENTER

The menu in Figure F-16 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

**FIGURE F-16. External Pressure Parameters Menu**

```
------ Pressure (current value in [])-------
     1) $PRESSURE_YINT [-43.606201] Pressure, Y-intercept (psig)
     2) $PRESSURE_SLOPE [0.00011645618] Pressure, slope
        (psig/AD$)                                  I
     3) $AD7714Ch0Gain [128] AD7714 pressure chan gain
        (1=simulation/64=Druck 4020/128=Paine)
    CR) Return to previous menu
Enter selection (1-3,CR): |
```

Initial set-up of the External Pressure Parameters is done at the factory. $Pressure_YINT is updated by Seaglider during self test. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

- To exit the External Pressure Parameters menu and return to the Parameters and Configuration menu, press ENTER.

## *Pressure (internal) Parameters*

To access the Internal Pressure Parameters menu from the Parameters and Configuration menu:

- Select **14**: Pressure (internal) Parameters and press ENTER

The menu in Figure F-17 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

**FIGURE F-17. Internal Pressure Parameters Menu**

```
------ IntPressure (current value in [])-------
     1) $INT_PRESSURE_SLOPE [0.0097660003] Internal Pressure, slope (psig/AD$)
     2) $INT_PRESSURE_YINT [0] Internal Pressure, y-intercept (psig)
    CR) Return to previous menu
Enter selection (1-2,CR):
```

Initial set-up of the Internal Pressure Parameters is done at the factory. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

- To exit the Internal Pressure Parameters menu and return to the Parameters and Configuration menu, press ENTER.

## *Compass Parameters*

To access the Compass Parameters menu from the Parameters and Configuration menu:

- Select **15**: Compass Parameters and press ENTER

The menu in Figure F-18 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name

**FIGURE F-18. Compass Parameters Menu**

```
------ Compass (current value in [])-------
          1) $TCM_PITCH_OFFSET [0] Pitch sensor offset (deg)
          2) $TCM_ROLL_OFFSET [0] Roll sensor offset (deg)
          3) $COMPASS_USE [0] Flags to indicate trust in compass returns
         CR) Return to previous menu
 Enter selection (1-3,CR):
```

Compass calibration is done at the factory. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

- To exit the Compass Parameters menu and return to the Parameters and Configuration menu press, ENTER.

## *Altimetry Parameters*

To access the Altimetry Parameters menu from the Parameters and Configuration menu:

- Select **16**: Altimetry Parameters and press ENTER

The menu in Figure F-19 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name. The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83.

**FIGURE F-19. Altimetry Parameters Menu**

```
------ Altimeter (current value in [])-------
         1) $ALTIM_BOTTOM_PING_RANGE [0] Range from (presumed) apogee
          depth to initiate bottom pings (m)
         2) $ALTIM_TOP_PING_RANGE [0] Depth to initiate top pings (m)
         3) $ALTIM_BOTTOM_TURN_MARGIN [0] Distance from obstacle to
          initiate bottom turn (m)
         4) $ALTIM_TOP_TURN_MARGIN [0] Distance from obstacle to initiate
            top turn (m)
         5) $ALTIM_TOP_MIN_OBSTACLE [1] Minimum obstacle depth to honor
            in initiating a subsurface finish (m)
         6) $ALTIM_PING_DEPTH [80] Depth at which to begin pinging for
          bottom (m)
         7) $ALTIM_PING_DELTA [5] Depth increment when pinging for bottom
          (m)
         8) $ALTIM_FREQUENCY [13] Frequency to use for pings (kHz)
         9) $ALTIM_PULSE [3] Pulse length to use for pings (ms)
         10) $ALTIM_SENSITIVITY [3] Altimeter sensitivity (0-5)
         11) $XPDR_VALID [2] Transponder interrogate validity length (x0.5
          ms) (0-6)
         12) $XPDR_INHIBIT [90] Transponder inhibit length (x100 of ms)
          (0-99)
          CR) Return to previous menu
Enter selection (1-12,CR):
```

To change a parameter value, follow the procedures “Basic Mission and Seaglider Parameters” on page 371.

## *Sea-Bird CT Calibration Coefficients*

To access the SBE CT Calibration Coefficients menu from the Parameters and Configuration menu:

- Select **17**: SBE CT Coefficients Menu and press ENTER

The menu in Figure F-20 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

**FIGURE F-20. Sea-Bird CT Calibration Coefficients Menu**

```
------ Parameters (current value in [])-------
     1) $SEABIRD_T_G [0.004302992] Seabird T g coefficient
     2) $SEABIRD_T_H [0.00062119117] Seabird T h coefficient
     3) $SEABIRD_T_I [2.2739867e-05] Seabird T i coefficient
     4) $SEABIRD_T_J [2.4046374e-06] Seabird T j coefficient
     5) $SEABIRD_C_G [-9.8404341] Seabird C g coefficient
     6) $SEABIRD_C_H [1.104722] Seabird C h coefficient
     7) $SEABIRD_C_I [-0.0022409149] Seabird C i coefficient
     8) $SEABIRD_C_J [0.0002592774?] Seabird C j coefficient
    CR) Return to previous menu
Enter selection (1-8,CR):
```

The Sea-Bird CT calibration coefficients are loaded into Seaglider at the factory. The coefficients are CT sensor specific. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

**Note:** For units with GPCTD installed or no CT Sail, these parameters will be set to 0.

**Note:** If changes are made to a SEABIRD coefficient, the change should also be made in that Seaglider's sg_calib_constants.m file. See Appendix B, "Seaglider File Formats Manual" on page 275.

- To exit the Sea-Bird CT Calibration Coefficients menu and return to the Parameters and Configuration menu, press ENTER.

## *Power Parameters*

To access the Power Parameters menu from the Parameters and Configuration menu:

- Select **18**: Power Parameters and press ENTER

The menu in Figure F-21 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name

**FIGURE F-21. Power Parameters Menu**

```
Enter selection (1-27,CR): 18
------ Power (current value in [])-------
          1) $AH0_24V [150] 24V-pack capacity (AmpHr)
          2) $AH0_10V [100] 10V-pack capacity (AmpHr)
          3) $MINV_24V [19] 24V-pack minimum voltage
          4) $MINV_10V [8] 10V-pack minimum voltage
          5) $FG_AHR_10V [0] 10V fuel gauge consumption
          6) $FG_AHR_24V [0] 24V fuel gauge consumption
          7) $PHONE_SUPPLY [2] Specifies battery for Iridium modem
           (1=10V, 2=24V)
         CR) Return to previous menu
Enter selection (1-7,CR): |
```

The Power Parameters are loaded into Seaglider at the factory. It is not recommended that the user change any of the values in this menu without consulting iRobot Customer Service.

## *Edit All Parameters*

To access the Edit All Parameters menu from the Parameters and Configuration menu:

- Select **19**: Edit All Parameters and press ENTER

One hundred sixty-nine Seaglider parameters, with the current value assigned in brackets, are listed. Figure F-22. To edit a parameter value, follow the procedures “Basic Mission and Seaglider Parameters” on page 371.

**FIGURE F-22. Edit All Parameters Menu**

```
|-----. All (current value in [])-------
  1) $ID_[169] Glider ID (3 digits max)
  2) $MISSION_[9] Current mission number
  3) $DIVE_[2] Next dive number
  4) $D_FLARE_[3] Flare depth (m)
  5) $D_TGT_[75] Target depth (m)
  6) $D_ABORT_[1000] Abort depth (m)
  7) $D_NO_BLEED_[200] Depth below which we will not bleed on dives (m)
  8) $D_BOOST_[0] Depth above which only boost pump is run (m)
  9) $T_BOOST_[0] Time to run the boost pump before turning off (0 do not turn off)
 10) $D_FINISH_[0] Dive finish depth (m)
 11) $D_PITCH_[0] Depth to achieve before surface maneuver pitch (m)
 12) $D_SAFE_[0] Safety dive depth for emergency/escape maneuvers (m)
 13) $D_CALL_[0] Depth to achieve before antenna is assumed to be out of water (m)
 14) $SURFACE_URGENCY_[0] Number of dives to accumulate before trying extra surfacings
 15) $SURFACE_URGENCY_FORCE_[0] Dive number modulo for forced extra surfacing attempts
 16) $T_DIVE_[42] Target time per dive/climb cycle (min)
 17) $T_MISSION_[55] Maximum mission time per dive/climb cycle (min)
 18) $T_ABORT_[1440] Failsafe watchdog  mission time - change with extreme care (min)
 19) $T_TURN_[225] Max time allowed for a turn maneuver (sec)
 20) $T_TURN_SAMPINT_[5] Sample interval during a turn (int sec, >=4)
 21) $T_NO_W_[120] Time with no vertical velocity (w) before taking next step (secs)
 22) $T_LOITER_[0] Time to loiter at apogee before climbing (secs)
 23) $USE_BATHY_[0] use bathymetry map (0 - no, +/-nnn - file number)
 24) $USE_ICE_[0] use ice map for surfacing decisions (0 - no, +/-nnn - file number)
 25) $ICEFREE_LON_[0.000000] Margin (degC) in freezing point check for surfacing
 26) $T_WATCHDOG_[10] Watchdog timer (min) outside bathymap (m)
 27) $REALAUNCH_[1] Relaunch glider if crash to TOMS? (0=no/1=yes)
 28) $APOGEE_PITCH_[-9] Intermediate pitch angle @ apogee (deg)
 29) $MAX_BUOY_[200] Max buoyancy force allowed @ apogee (g)
 30) $COURSE_BIAS_[0] Course bias to subtract from the computed (deg, +cw)
 31) $GLIDE_SLOPE_[30] Max glide slope allowed (deg)
 32) $SPEED_FACTOR_[1] Fudge factor for glider speed
 33) $RHO_[1.0233] Water density (grams/cc)
 34) $MASS_[53733] Glider mass (g)
 35) $NAV_MODE_[5] Navigation mode (0=direct/1=kalman/2=Set correct/3=Current relative)
 36) $KALMAN_USE_[0] kalmy.ma file (0=create/1=correct/2=navigation (deg)
 37) $HD_A_[0.003] Glider hydrodynamic lift (A)
 38) $HD_B_[0.0142] Glider hydrodynamic drag (B)
 39) $HD_C_[9.9999997e-06] Glider hydrodynamic induced drag (C)
 40) $HEADING_[-1] Heading (degrees true) to steer (-1 disable)
 41) $ESCAPE_HEADING_[0] Escape heading (degrees magnetic) to steer in emergency
 42) $ESCAPE_HEADING_DELTA_[10] Escape heading offset (degrees) in emergency
 43) $FIX_MISSING_TIMEOUT_[0] Time (days) without fix before activating escape heading
 44) $TGT_DEFAULT_LAT_[4743.75] Default target latitude
 45) $TGT_DEFAULT_LON_[-12224.15] Default target longitude
 46) $TGT_DEFAULT_NAME_[0] Automatically update default target
 47) $MAX_FILE_SIZE_[0] Max file size for upload (KBytes <=16, 0=no split, negative=no gzip)
 48) $NUM_FILES_[4] File management: surfacings per file (0=none, 1=only store compressed, 2=delete splits on failed phone call
 49) $FILEMGR_[0] File management aggressiveness: 0-none, 1-only store compressed, 2-delete splits on failed phone call
 50) $CALL_NDIVES_[1] # of dive/climb cycles before calling
 51) $COMM_SEQ_[0] Comm sequence to use (0-default, 1-ctl only, 2-no loggers)
 52) $KERMIT_[0] use of kermit (0 - none, 1 - data only, 2 - data only, batch, 3 data and control, 4 data and control - data batch)
 53) $N_NOCOMM_[1] # dives without communications before max surface
 54) $N_NOSURFACE_[0] modulo for dives to finish under surface
 55) $UPLOAD_DIVES_MAX_[-1] maximum # of dives to upload at each surfacing (-1 = all available)
 56) $CALL_TRIES_[5] # of calls to try at each surfacing
```

```
611) $CALL_WAIT [60] Delay between phone tries (sec)
612) $CAPUPLOAD [1] Upload capture file (0=no/1=yes)
613) $CAPMAXSIZE [100000] Maximum capture file size to upload (before compression)
614) $HEAPDBG [0] Turn heap debugging (0=off/1=on)
615) $N_GPS [20] Max # invalid GPS readings
616) $T_GPS_ALMANAC [0] Time to wait to reacquire the GPS almanac (mins)
617) $T_GPS_CHARGE [-14853.346] How long to trickle charge the GPS (secs)
618) $T_RSLEEP [3] Sleep interval during recovery (min)
619) $STROBE [0] Strobe control (0=never, 1=recovery, 2=surface)
620) $RAFOS_PEAK_OFFSET [1.5] Internal delay of RAFOS receiver (seconds)
621) $RAFOS_CORR_THRESH [60] threshold correlation value for hits used in navigation
622) $RAFOS_HIT_WINDOW [3600] window length (seconds) in which to search for hits to use in navigation
623) $PITCH_MIN [160 (-8.1 cm)] Pitch position limit, fwd (AD#)
624) $PITCH_MAX [3940 (3.7 cm)] Pitch position limit, aft (AD#)
625) $PITCH_CNV [-0.0031257631] Pitch position to cm conversion factor (cm/AD#)
626) $PITCH_DBAND [0.03999999] Pitch mass overshoot after motor off (cm)
627) $PITCH_GAIN [29] Pitch angle to mass gain (deg/cm)
628) $PITCH_TIMEOUT [16] Pitch mass timeout (sec)
629) $PITCH_AD_RATE [175] Pitch motor AD counts per sec
630) $PITCH_MAXERRORS [1] Number of pitch errors allowed before entering recovery
631) $PITCH_AD_GAIN [0] Gain for auto-adjusting pitch (cm/deg)
632) $PITCH_AD_DBAND [0] Deadband for auto-adjusting pitch (deg)
633) $ROLL_MIN [1877 (-37.6 deg)] Roll position limit, port (AD#)
634) $ROLL_MAX [2873 (46.9 deg)] Roll position limit, stbd (AD#)
635) $ROLL_DEG_LIMIT [2951] position normal flight (deg) and testing, (AD#)
636) $ROLL_CLIMB [2180] Roll neutral point when climbing (AD#)
637) $HEAD_ERRBAND [+/- 10.00] Roll (scene)/Heading (auto) deadband (deg)
638) $ROLL_CNV [0.02827001] Roll position conversion factor (deg/AD#)
639) $ROLL_TIMEOUT [15] Roll mass timeout, (roll and rollback, sec)
640) $R_PORT_OVSHOOT [50] Roll mass overshoot to port after motor off (AD)
641) $R_STBD_OVSHOOT [51] Roll mass overshoot to stbd after motor off (AD)
642) $ROLL_AD_RATE [350] Roll motor AD count per sec
643) $ROLL_MAXERRORS [1] Number of roll errors allowed before entering recovery
644) $ROLL_ADJ_GAIN [0] Gain for auto-adjusting roll center (deg battery roll / deg/sec turn rate
645) $ROLL_ADJ_DBAND [0.5] Deadband for auto-adjusting roll center (deg/sec)
646) $VBD_MIN [3060 (5.5212 cc)] VBD bladder position limit, minimum (AD$)
647) $VBD_MAX [30751] VBD bladder position limit, maximum (AD$)
648) $CVBD_PARK [30751] VBD bladder neutral point (AD$)
649) $VBD_DBAND [+/- 2.00] VBD bladder position deadband (cc)
650) $VBD_CNV [-0.245296] VBD bladder position conversion factor (cc/AD$)
651) $VBD_TIMEOUT [720] VBD timeout (sec)
652) $VBD_VBD_SHIFT [0.001230000l] Pitch offset due to VBD (cm/cc)
653) $VBD_PUMP_AD_RATE_SURFACE [5] VBD pump AD counts per sec at surface
654) $VBD_PUMP_AD_RATE_APOGEE [4] VBD pump AD counts per sec at apogee
655) $VBD_BLEED_AD_RATE [8] VBD bleed AD counts per sec at surface
656) $UNCOM_BLEED [20] Uncommanded bleed change (AD) 0 - disabled
657) $VBD_MAXERRORS [2] Number of VBD errors allowed before entering recovery
658) $AH0_24V_ERRORS [20] Number of EFB errors allowed before entering recovery
659) $AH0_10V [1000] 10v-pack capacity (AmpHr)
660) $MINV_24V [19] 24v-pack minimum voltage
661) $MINV_10V [8] 10v-pack minimum voltage
662) $FG_AHR_10V [0] 10v fuel gauge consumption
663) $FG_AHR_24V [0] 24v fuel gauge consumption
664) $PHONE_SUPPLY [2] Specifies battery for Iridium modem (1=10V, 2=24V)
665) $PRESSURE_YINT [-44.96897l] Pressure, y-intercept (psig)
666) $PRESSURE_SLOPE [0.000116456l8] Pressure, slope (psig/AD$)
```

```
122) $AD7714CH0Gain [128] AD7714 pressure chan gain (1=simulation/64=Druck 4020/128=Paine)
123) $TCM_PITCH_OFFSET [0] Pitch sensor offset (deg)
124) $TCM_ROLL_OFFSET [0] Roll sensor offset (deg)
125) $TCM_TRUST [0] Trust compass returns
126) $ALTIM_BOTTOM_PING_RANGE [0] Initiate bottom pings (m) approx depth to initiate bottom pings (m)
127) $ALTIM_TOP_PING_RANGE [0] Depth to initiate top pings (m)
128) $ALTIM_BOTTOM_TURN_MARGIN [12] Distance from obstacle to initiate bottom turn (m)
129) $ALTIM_TOP_TURN_MARGIN [0] Distance from obstacle to initiate top turn (m)
130) $ALTIM_TOP_MIN_OBSTACLE [1] Minimum obstacle depth to honor in initiating a subsurface finish (m)
131) $ALTIM_PING_DEPTH [80] Depth at which to begin pinging for bottom (m)
132) $ALTIM_PING_DELTA [5] Depth increment when pinging for bottom (m)
133) $ALTIM_FREQUENCY [13] Frequency to use for pings (kHz)
134) $ALTIM_PULSE [3] Pulse length to use for pings (ms)
135) $ALTIM_SENSITIVITY [3] Altimeter sensitivity (0-5)
136) $XPDR_VALID [2] Transponder interrogate validity width (x0.5 ms) (0-6)
137) $XPDR_SLOPE [0.0097600003] Internal Pressure slope (psig/AD$)
138) $INT_PRESSURE_YINT [2.31] Internal Pressure y-intercept (psig)
139) $INT_PRESSURE_SLOPE [0.0097600003] Internal Pressure slope (psig/AD$)
140) $DEEPGLIDER [0] Am I a Deep Glider (0=No, 1 = Yes)
141) $DEEPGLIDERMB [0] Motherboard stuffed with high current sensors (0=No, 1 = Yes)
142) $MOTHERBOARD [4] Motherboard revision (NOPP=1,A.1=2,B.1=3,C=5)
143) $DEVICE1 [-1] Attached device 1 flags
144) $DEVICE2 [-1] Attached device 2 flags
145) $DEVICE3 [-1] Attached device 3 flags
146) $DEVICE4 [-1] Attached device 4 flags
147) $DEVICE5 [-1] Attached device 5 flags
148) $DEVICE6 [-1] Attached device 6 flags
149) $DEVICES [-1] Attached devices enable control
150) $LOGGERDEVICE1 [85] Attached logger device 1 flags
151) $LOGGERDEVICE2 [-1] Attached logger device 2 flags
152) $LOGGERDEVICE3 [0] Attached logger device 3 flags
153) $LOGGERDEVICE4 [-1] Attached logger device 4 flags
154) $COMPASS_DEVICE [33] Compass device flags
155) $COMPASS2_DEVICE [-1] Compass 2 device flags
156) $PHONE_DEVICE [48] Phone device flags
157) $GPS_DEVICE [3] GPS flags
158) $RAFOS_DEVICE [-1] RAFOS device flags
159) $XPDR_DEVICE [24] xPDR device flags
160) $SIM_PRESSURE [0] Simulated pressure (<0 use pressure gauge to compute depth
161) $SIM_PITCH [0] Simulated pitch (degrees) 0 - use TCM2 to report pitch
162) $SEABIRD_T_G [0.0004302892] Seabird T (g) coefficient
163) $SEABIRD_T_H [0.0006211911] Seabird T h coefficient
164) $SEABIRD_T_I [2.2739867e-05] Seabird T i coefficient
165) $SEABIRD_T_J [2.4046374e-06] Seabird T j coefficient
166) $SEABIRD_C_G [-9.8404341] Seabird C g coefficient
167) $SEABIRD_C_H [1.1047221] Seabird C h coefficient
168) $SEABIRD_C_I [-9.9999997e-05] Seabird C i coefficient
169) $SEABIRD_C_J [0.0002592747] Seabird C j coefficient
  CR) Return to previous menu
Enter selection (1-169,CR): |
```

**Note:** Changes made to parameters in this section can also be made in the cmdfile. To edit the cmdfile go to “Command File” on page 239 for an explanation. Any edits made to the SEABIRD coefficients either through this menu or the cmdfile should also be made in the sg_calib_constants.m file. (See Appendix B, “Seaglider File Formats Manual” on page 275 for information on this file).

## *Validate Parameters*

To access the Validate Parameters menu from the Parameters and Configuration menu:

- Select **20**: Validate Parameters and press ENTER

The parameters are written to NVRAM.

When the write is complete the user is automatically returned to the Parameters and Configuration menu.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Show Parameter Details*

To access the Validate Parameters menu from the Parameters and Configuration menu:

- Select **21**: Show Parameters and press ENTER

A table listing all Seaglider parameters, the group each parameter is in (basic, dive, surface, flight etc.) the nominal, min and max values and a brief definition is displayed. No edits may be made in this menu.

After the table is displayed the user is automatically returned to the Parameters and Configuration menu.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Show Changed Parameters*

To access the Show Changed Parameters Menu from the Parameters and Configuration menu:

- Select **22**: Show Changed Parameters and press ENTER

The names of the parameters whose values were changed under menu option 19 and the respective new parameter value are displayed.

After the changed parameters are listed the user is automatically returned to the Parameters and Configuration menu.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Clear Changed Parameters*

To access the Clear Changed Parameters menu from the Parameters and Configuration menu:

- Select **23**: Clear Changed Parameters and press ENTER

This option returns any parameter values changed under option 19: Edit All Parameters to the original value.

After all parameter values are reverted back to original value the Parameters and Configuration menu is displayed.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Save Parameters By Name To A File*

To access the Save Parameters By Name To A File option from the Parameters and Configuration menu:

- Select **24**: Save Parameters By Name To A File and press ENTER

The parameters and their respective current values are written to a file on the computer connected to Seaglider. Figure F-23. The filename format is SGxxxPRM.TXT where xxx is Seaglider's ID number.

**FIGURE F-23. Save Parameters By Name To A File**

```
13773.521,SUSR,N,Saving to SG506PRM.TXT...|
```

After the file is written the Parameters and Configuration menu is displayed.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Dump Parameters To Screen*

To access the Dump Parameters To Screen option from the Parameters and Configuration menu:

- Select **25**: Dump Parameters To Screen and press ENTER

All of the parameters and their respective current values are written to the screen.

After the file is written the Parameters and Configuration menu is displayed.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Load Parameters From A File*

To access the Load Parameters From A File option from the Parameters and Configuration menu:

- Select **26**: Load Parameters From A File and press ENTER

The file SGxxxPRM.TXT is loaded from the laptop onto Seaglider. (substitute xxx for Seaglider's ID number)

After the file is written the Parameters and Configuration menu is displayed.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Reset To Defaults*

To access the Reset to Defaults option from the Parameters and Configuration menu:

- Select **27**: Reset to Defaults and press ENTER

The default value for each parameter is loaded.

After the file is written the Parameters and Configuration menu is displayed.

- To exit from the Parameters and Configuration menu and return to the Main Menu, press ENTER.

## *Hardware Menu (for Tests and Monitoring)*

To view the Hardware Menu fused or tests and monitoring, Figure F-24:

- Select **2** and press ENTER.

**FIGURE F-24. Hardware Menu**

```
------ Hardware Menu ------
*Motors and VBD
     1 [pitch   ]  Pitch control
     2 [roll    ]  Roll control
     3 [vbd     ]  VBD control
*Fixed Devices
     4 [super   ]  Supervisor
     5 [pressure]  Pressure sensor
     6 [compass ]  Compass (tcm2)
     7 [gps     ]  GPS
     8 [modem   ]  Modem (xmodem mode)
     9 [intpress]  Internal pressure
    10 [altim   ]  Altimeter
    11 [sensors ]  Sensors
    12 [loggers ]  Loggers
*Other
    13 [batt    ]  Batteries and fuel gauges
    14 [lowlevel]  Low-level hardware (IO,A-D,CF)
    15 [misc    ]  Miscellaneous (travel, timeouts, date/time)
    16 [develop ]  Developer tests
    CR)  Return to previous
```

## *Pitch Control Menu*

To access the Pitch Control Menu:

- Select **1**: Pitch Control and press ENTER

The menu in Figure F-25 is displayed.

**FIGURE F-25. Pitch Control Menu**

```
------ Pitch Control Menu ------
        1 [read    ]  Current position
        2 [ad      ]  Move to position (AD counts)
        3 [eu      ]  Move to position (cm, - for fwd)
        4 [edit    ]  Edit pitch parameters
        5 [cycle   ]  Run pitch duty cycles
        6 [cyclepr ]  Run pitch & roll duty cycles (CTL-Q to quit)
        7 [pitch   ]  Pitch test
       CR)  Return to previous
```

## *Current Motor Position*

To determine the present position of the pitch motor:

- Select **1**: Current Position and press ENTER

A reply similar to the one below is returned. Note that the pitch control position is given both in cm and A/D counts.

```
15313.427,HPITCH,N,   Pitch control position = -8.80 cm,
         (AD#= 365 )|
                                                    T
```

The user is returned to the Pitch Control Menu.

## *Move to Position (AD counts and cm)*

There are two options to move the pitch motor:

- Change the A/D counts.
- Change the cm (centimeters) of movement.

### Change the A/D counts

To use this method:

- Select **2**: Move to Position (AD counts) and press ENTER.

Seaglider responds with:

```
15582.616,HPITCH,N,   Pitch control position = -8.74 cm, (AD#= 385 )
Move to AD#:  [385]|
```

Type an A/D value and press ENTER. Below is an example showing and A/D count change to 395 and the resulting motor movement. Figure F-26.

**FIGURE F-26. Pitch Motor Movement By Changing A/D Counts**

```
Move to AD#:  [385] 395
15599.680,HPITCH,N,Pitch commanded from -8.74 cm (385) to -
        8.71 cm (395)...
15600.932,HPITCH,N,-8.7 cm (ad: 385) MOTOR_DONE: ticks: 24
max 24v: 4.6mA avg 24v: 3.8mA minV 24v: 23.6V
15601.402,SMOTOR,N,GC TICKS/TIME: 27/28663
15601.492,HPITCH,N,done.
```

At the end of the motor movement the user is returned to the Pitch Control Menu.

### Change the cm (centimeters) of movement

To use this method:

1. Select **3**: Move to Position (cm) and press ENTER.

   The response to this command is the same as the response to changing the A/D counts.

2. Type a cm value and press ENTER.

   A positive value moves the pitch motor towards the aft end of Seaglider, a negative value moves the pitch motor towards the nose. Seaglider outputs the resulting movement like it did for the A/D count change above.

   At the end of the motor movement the user is returned to the Pitch Control Menu.

## *Edit Pitch Parameters*

To access the Pitch Parameters menu for editing:

- Select **4**: Edit Pitch Parameters and press ENTER

The menu shown in Figure F-12 is displayed. The present value for each parameter is shown in brackets to the right of the parameter name.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $PITCH_MIN, $PITCH_MAX and $C_PITCH that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

The instructions for editing a pitch parameter are located under “Pitch Parameters” on page 378.

- To exit this menu and return to the Pitch Control Menu, press ENTER.

## *Run Pitch Duty Cycles*

To run pitch duty cycles:

- Select **5**: Run Pitch Duty Cycles and press ENTER.

As an example, Seaglider responds with:

```
Enter positions between -9.4 and 2.4 cm:
Start position of cycle: [-9.439804]
```

As an example  enter -8.5 and press ENTER.

```
End position of cycle: [2.375580] |
```

As an example enter 1.5 and press ENTER.

```
# of secs to rest at end of each cycle: [1]|
```

As an example use the default so press ENTER.

```
# of cycles to execute (<32768): [10]|
```

As an example, enter **1** and press ENTER.

Seaglider then cycles the pitch motor and outputs the movement to the screen as the exercise is happening. Figure F-27.

To abort the test at any time press any key on the keyboard.

**FIGURE F-27. Pitch Duty Cycling**

```
Cycling between -8.50 and 1.50 cm, hit any key to abort!
15875.935,HPITCH,N,Pitch commanded from -8.74 cm (385) to -8.50 cm (460)...
15877.053,HPITCH,N,-8.5 cm (ad: 449) MOTOR_DONE: ticks: 8 max 24v: 3.8mA avg 24v: 3.1mA minV 24v: 23.5V
15877.551,SMOTOR,N,GC TICKS/TIME: 26/28391
15877.641,HPITCH,N,done.
15877.783,HPITCH,N,Pitch completed from -8.74 cm (385) to -8.54 cm (449) took 0.4 sec 284mA (302mA peak) 23.1Vmin 170 AD/sec 15
15878.176,HPITCH,N,TRACK: b: 446/0 a: 448/0 d: -1 o: 1
15878.403,HPITCH,N,Pitch commanded from -8.54 cm (449) to 1.50 cm (3659)...
15879.585,HPITCH,N,-8.3 cm (ad: 527)
15880.335,HPITCH,N,-7.9 cm (ad: 668)
15881.110,HPITCH,N,-7.4 cm (ad: 815)
15881.860,HPITCH,N,-7.0 cm (ad: 953)
15882.635,HPITCH,N,-6.5 cm (ad: 1094)
15883.410,HPITCH,N,-6.1 cm (ad: 1234)
15884.160,HPITCH,N,-5.7 cm (ad: 1371)
15884.935,HPITCH,N,-5.2 cm (ad: 1513)
15885.710,HPITCH,N,-4.8 cm (ad: 1657)
15886.460,HPITCH,N,-4.3 cm (ad: 1796)
15887.210,HPITCH,N,-3.9 cm (ad: 1938)
15887.985,HPITCH,N,-3.5 cm (ad: 2075)
15888.760,HPITCH,N,-3.0 cm (ad: 2215)
15889.510,HPITCH,N,-2.6 cm (ad: 2352)
15890.285,HPITCH,N,-2.2 cm (ad: 2491)
15891.035,HPITCH,N,-1.7 cm (ad: 2626)
15891.785,HPITCH,N,-1.3 cm (ad: 2765)
15892.560,HPITCH,N,-0.9 cm (ad: 2895)
15893.310,HPITCH,N,-0.5 cm (ad: 3029)
15894.060,HPITCH,N,-0.1 cm (ad: 3159)
15894.835,HPITCH,N,0.3 cm (ad: 3288)
15895.585,HPITCH,N,0.8 cm (ad: 3422) Pitch motor timeout!
15896.235,HPITCH,N,
15896.385,HPITCH,N,1.1 cm (ad: 3521) MOTOR_DONE: ticks: 17 max 24v: 3.8mA avg 24v: 3.8mA minV 24v: 23.5V
15896.880,SMOTOR,N,GC TICKS/TIME: 700/702177
15896.972,HPITCH,N,done.
15897.111,HPITCH,C,Pitch completed from -8.54 cm (449) to 1.07 cm (3521) took 17.0 sec 273mA (304mA peak) 23.1Vmin 180 AD/sec 68
15897.541,HPITCH,N,TRACK: b: 3511/0 a: 3511/0 d: 135 o: 10
15898.284,HPITCH,N,Pitch commanded from 1.06 cm (3520) to -8.50 cm (460)...
15899.494,HPITCH,N,0.7 cm (ad: 3391)
15900.244,HPITCH,N,-0.1 cm (ad: 3161)
15900.994,HPITCH,N,-0.8 cm (ad: 2929)
15901.744,HPITCH,N,-1.5 cm (ad: 2687)
15902.519,HPITCH,N,-2.3 cm (ad: 2449)
15903.269,HPITCH,N,-3.0 cm (ad: 2210)
15904.019,HPITCH,N,-3.8 cm (ad: 1962)
15904.794,HPITCH,N,-4.6 cm (ad: 1722)
15905.544,HPITCH,N,-5.3 cm (ad: 1475)
15906.319,HPITCH,N,-6.1 cm (ad: 1236)
15907.094,HPITCH,N,-6.8 cm (ad: 991)
15907.844,HPITCH,N,-7.6 cm (ad: 755)
15908.594,HPITCH,N,-8.3 cm (ad: 511) MOTOR_DONE: ticks: 2 max 24v: 3.8mA avg 24v: 3.8mA minV 24v: 23.5V
15909.137,SMOTOR,N,GC TICKS/TIME: 394/396216
15909.229,HPITCH,N,done.
15909.369,HPITCH,N,Pitch completed from 1.07 cm (3521) to -8.50 cm (462) took 9.7 sec 79mA (321mA peak) 23.4Vmin 314 AD/sec 389
15909.765,HPITCH,N,TRACK: b: 473/0 a: 469/0 d: 3 o: -7
Cycle 1 completed...
```

At the end of the motor movement the user is returned to the Pitch Control Menu.

## *Run Pitch and Roll Duty Cycles*

To run pitch and roll duty cycles at the same time:

- Select **6**: Run Pitch and Roll Duty Cycles and press ENTER.

Seaglider responds with the limits within which pitch and roll can be moved and the format for the desired duty cycles.

```
Pitch limits: -9.4 and 2.4 cm for pitch.
Roll limits: -59.6 and 44.8 deg for roll.
Enter moves as p,cm r,deg or s,milliseconds, one per line
Ctrl-C aborts, blank line terminates entry
```

An example of user input for a pitch and roll duty cycle is:

```
0: p,-8
1: r,-3
2: p,-6
3: s,5
4:      |
```

at this step, input is complete so press ENTER

Seaglider responds with what it thinks it heard:

```
p,-8.000000
r,-35.000000
p,-6.000000
s,5.000000
```

and asks for the number of cycles

```
# of cycles to execute: [10] 1|
```

Seaglider cycles the pitch and roll motors and outputs the movement to the screen as the exercise is happening.

To abort the test at any time, press any key on the keyboard.

At the end of the motor movement the user is returned to the Pitch Control Menu.

## *Pitch Test*

To run the pitch motor from software stop to stop without user input:

- Select **7**: Pitch Test and press ENTER

Seaglider then outputs to the screen the pitch motor movement.

At the conclusion of the test the user is returned to the Pitch Control Menu.

To return to the Hardware Menu, one level up, press ENTER.

## *Roll Control Menu*

To access the Roll control menu:

- Select **2**: Roll Control and press ENTER

The menu in Figure F-28 is displayed.

**FIGURE F-28. Roll Control Menu**

```
------ Roll control menu ------
     1 [read    ]  Current position
     2 [ad      ]  Move to position (AD counts)
     3 [eu      ]  Move to position (deg, + for stbd)
     4 [edit    ]  Edit roll parameters
     5 [cycle   ]  Run roll duty cycles
     6 [cycle pr]  Run pitch & roll duty cycles (CTL-Q to quit)
     7 [rolls   ]  Roll test
    CR)  Return to previous
```

The method for accessing these options and performing tests is the same as it is for Pitch Control discussed in “Pitch Parameters” on page 378.

The instructions for making edits to the roll parameters are located in “Roll Parameters” on page 379.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $ROLL_MIN, $ROLL_MAX and $C_ROLL that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

**Note:** When making roll movements in degrees, negative degree values roll the vehicle to port and positive degree values roll the vehicle to starboard.

To return to the Hardware Menu, one level up, press ENTER.

## *VBD Menu*

To access the VBD menu:

- Select **3**: VBD Control and press ENTER

The menu in Figure F-29 is displayed.

**FIGURE F-29. VBD Control Menu**

```
------ VBD menu ------
     1 [read    ]  Current position
     2 [ad      ]  Move to position (AD counts)
     3 [eu      ]  Move to position (cc)
     4 [param   ]  Edit vbd parameters
     5 [timing  ]  Characterize valve
     6 [noise   ]  Valve noise test
     7 [open    ]  Open valve
     8 [close   ]  Close valve
     9 [cycle   ]  Cycle valve
    10 [pump    ]  Pump & bleed cycles (pressure chamber or AD)
    11 [soak    ]  Pump and hold at pressure (pressure chamber or AD)
    12 [special ]  Special test #1 (motor current, amb pressure & pot)
    CR)  Return to previous
```

The method for accessing options 1-4 is the same as it is for Pitch Control in “Pitch Control Menu” on page 394.

The instructions for making edits to the VBD parameters are located in “VBD Parameters” on page 380.

The explanation for each parameter as well as the acceptable range of values is located in Chapter 5, “Piloting Parameters” on page 83. The exact values of $VBD_MIN, $VBD_MAX and $C_VBD that should be used for the Seaglider being queried are located on the trim sheets in the notebook shipped with Seaglider.

**Note:** When making VBD moves in AD counts, increasing the count moves oil into the internal reservoir. When making VBD moves in cc, negative values mean oil is moving into the internal reservoir.

Options 5-12 are used for Seaglider development. Most users will never have a need to use any of these options. Should the user be interested, brief discussions of these options follow.

## *Characterize Valve*

To characterize the valve:

- Select **5**: Characterize Valve and press ENTER

Seaglider responds with:

```
Delay in seconds on bleeds before checking progress: [2] |
```

If a 2 second delay (the default) is acceptable press ENTER. Otherwise type in a new value and press ENTER.

```
VBD change assumed after how many counts? [2] |
```

If 2 counts (the default) is acceptable press ENTER. Otherwise type in a new value and press ENTER.

```
47 Valve opened at 0 ms.
.45 .44 .44 .43 .42 .42 .43 .0 Valve closed at 7 ms.|
```

Pump back to starting state? [Y]  Press ENTER to continue test. To stop the test type N and press ENTER.

Seaglider echoes the valve activity to the screen. Figure F-30.

**FIGURE F-30. VBD Valve Characterization**

```
19727.304,HVBD,N,Pump commanded from -122.65 cc (3656) to 10.55 cc (3113)...
19728.636,HVBD,N,-121.9 cc (ad: 3653 [3692, 3611])
19729.461,HVBD,N,-119.7 cc (ad: 3644 [3684, 3603])
19730.286,HVBD,N,-117.7 cc (ad: 3636 [3676, 3595]) .
20442.731,HVBD,N,-163.1 cc (ad: 3821 [3850, 3791])
20443.581,HVBD,N,-161.4 cc (ad: 3814 [3842, 3783])
20444.406,HVBD,N,-159.2 cc (ad: 3805 [3833, 3775])
20445.206,HVBD,N,-157.7 cc (ad: 3799 [3829, 3769]) MOTOR_DONE: ticks: 20 max 24v: 6.1mA avg 24v: 3.8mA minV 24v: 23.5v
20445.830,SMOTOR,N,GC TICKS/TIME: 1725/1730380
20445.925,HVBD,N,done.
20446.086,HVBD,N,Pump completed from -230.09 cc (4094) to -157.48 cc (3798 [3829, 3768]) took 37.5 sec 345mA (2623mA peak) 22.9Vmin 7 AD/sec 1501
20446.560,HVBD,N,TRACK: b: 3830/3770 a: 3831/3770 d: -31 o: -33|
```

At the conclusion of the valve characterization, Seaglider returns to the VBD menu.

## *Valve Noise Test*

To check the VBD valve noise:

- Select **6**: Valve Noise Test and press ENTER

Seaglider responds with:

```
Enter time in seconds to have the valve tested: [1] |
```

Press ENTER if default answer, 1 second, is okay or enter another value and press ENTER.

```
Open: Raise OPN before CLS? [Y] |
```

Press ENTER if default answer, Yes, is okay, if not enter N and press ENTER.

```
Close: Drop OPN before CLS? [Y] |
```

Press ENTER if default answer, Yes, is okay; if not enter N and press ENTER.

```
Offset time (in ms) between valve signals: [0] |
```

Press ENTER if default answer, 0, is okay or enter another value and press ENTER.

```
Bleed 4 counts on 1 second open and 21 counts on close.
Pump back to starting state? [Y] |
```

Press ENTER if default answer, Yes, is okay or enter N and press ENTER.

During the test, Seaglider echoes the output to the computer. Figure F-31.

**FIGURE F-31. Valve Noise Test**

```
19952.303,HVBD,N,Pump commanded from 4.17 cc (3139) to 11.04 cc (3111)...
19953.649,HVBD,N,4.9 cc (ad: 3136 [3181, 3090])
19954.449,HVBD,N,6.6 cc (ad: 3129 [3173, 3083])
19955.224,HVBD,N,8.6 cc (ad: 3121 [3165, 3075])
19956.049,HVBD,N,10.5 cc (ad: 3113 [3157, 3068]) MOTOR_DONE: ticks: 3 max 24v: 4.6mA avg 24v: 3.8mA minV 24v: 23.4V
19956.614,SMOTOR,N,GC TICKS/TIME: 127/129063
19956.706,HVBD,N,done.
19956.855,HVBD,N,Pump completed from 3.92 cc (3140) to 11.28 cc (3110 [3154, 3066]) took 3.0 sec 51mA (290mA peak) 23.4Vmin
19957.274,HVBD,N,TRACK: b: 3156/3067 a: 3156/3067 d: -45 o: -46
```

At the conclusion of the test, Seaglider returns to the VBD menu.

## *Open Valve*

To open VBD valve:

- Select 7: Open Valve and press ENTER

Seaglider returns to the VBD menu.

## *Close Valve*

To close VBD valve:

- Select **8**: Close Valve and press ENTER

Seaglider returns to the VBD menu.

## *Cycle Valve*

To cycle the VBD valve:

- Select **9**: Cycle Valve and press ENTER

Seaglider returns:

```
Enter time in ms to have the valve OPEN: [1000] |
```

If 1000ms is okay press ENTER otherwise type in a new value and press ENTER.

```
Number of cycles between reports: [20] 1|
```

If 1 is okay press ENTER otherwise type in a new value and press ENTER.

Seaglider then cycles the VBD valve echoing the activity back to the laptop. Figure F-32.

**FIGURE F-32. VBD Valve Cycling**

```
20147.873,HVBD,N,Hit any key to abort...
20149.171,HVBD,N,    VBD lin pot AD counts: 3178 3090
20149.315,HVBD,N,    VBD bladder position = 5.40 cc, (AD#= 3134 )
20150.611,HVBD,N,    VBD lin pot AD counts: 3202 3114
20150.756,HVBD,N,    VBD bladder position = -0.49 cc, (AD#= 3158 )
.
.
.
20209.852,HVBD,N,    VBD lin pot AD counts: 4077 4088
20210.007,HVBD,N,    VBD bladder position = -227.14 cc, (AD#= 4082)
20211.305,HVBD,N,    VBD lin pot AD counts: 4089 4094
20211.460,HVBD,N,    VBD bladder position = -229.35 cc, (AD#= 4091 )
20212.759,HVBD,N,    VBD lin pot AD counts: 4094 4094
20212.912,HVBD,N,    VBD bladder position = -230.09 cc, (AD#= 4094 )
20214.212,HVBD,N,    VBD lin pot AD counts: 4094 4095
20214.365,HVBD,N,    VBD bladder position = -230.09 cc, (AD#= 4094 )
```

At the conclusion of the test, Seaglider returns to the VBD menu.

## *Pump and Bleed Cycles*

This test is done in a pressure chamber. It is not recommended that this test be done outside of the iRobot factory.

To perform pump and bleed cycles:

- Select **10**: Pump and Bleed Cycles

Seaglider returns:

```
Do you want to specify pressure[Y] or VBD AD[N] limits? [Y] |
```

In this example pressure, the default, is the limit

```
20475.952,HVBD,N,
Current pressure = 0.1
Maximum pressure (psi): [1500.000000] 350|
```

Set the maximum pressure. For this example, pressure is set to 350

```
Time to rest @ max pressure (secs): [5] |
```

If 5 is okay press ENTER otherwise type in a new value and press ENTER.

```
Minimum pressure (psi): [0.000000]|
```

If 0 is okay press ENTER otherwise type in a new value and press ENTER.

```
Time to rest @ min pressure (secs): [5] |
```

If 5 is okay press ENTER otherwise type in a new value and press ENTER.

```
Sample report interval (secs): [1] |
```

If 1 is okay press ENTER otherwise type in a new value and press ENTER.

```
Display readings? [Y] |
```

If the readings should be displayed press ENTER otherwise type in a new value and press ENTER.

```
Number of cycles to run: [10] 1|
```

Set the number of cycles to run and press ENTER. For this example, the number of cycles is 1.

Seaglider starts the pump and bleed test and echoes the activity to the laptop screen. Figure F-33. To stop the test, press any key.

**FIGURE F-33. VBD Pump and Bleed Pressure Test**

```
20503.396,HVBD,N,Hit any key to abort!
20503.491,HVBD,N, cycle    sec  vbd0  vbd1   avg     mA        P   psi
20504.415,HVBD,N,1  Pressure start: 0  end: 1910  VBD start: 3798  end: 3797
   Time: 0  cc/sec: 0.00  avg mA: 0.000
20504.621,HVBD,N,----- pumped to     3828    3765    3796 (0)----
20504.747,HVBD,N,Waiting for 5 sec...
20510.299,HVBD,N,-----bled to     3834    3773    3803 (0)-----
20510.422,HVBD,N,Waiting for 5 sec...
20515.558,HVBD,N,Cycle 1 completed ...
20515.661,SPOWER,N,powerOFF (3,VBD_pump_during_surface) without corresponding p
```

At the conclusion of the test, Seaglider returns to the VBD menu.

## *Pump and Hold at Pressure*

This test is done in a pressure chamber. It is not recommended that this test be done outside of the iRobot factory.

To perform pump and hold at pressure cycles:

- Select **11**: Pump and Bleed Cycles

Seaglider returns:

```
Do you want to specify pressure[Y] or VBD AD[N] limits? [Y] |
```

In this example pressure, the default, is the limit.

```
20642.025,HVBD,N,
Current pressure = 0.2
Maximum pressure (psi): [1500.000000] .1|
```

Set the maximum pressure. For this example, pressure is set to .1

```
Sample report interval (secs): [1] |
```

If 1 is okay press ENTER otherwise type in a new value and press ENTER.

```
Display readings? [Y] |
```

If the readings should be displayed press ENTER otherwise type in a new value and press ENTER.

Seaglider starts the pump and hold test and echoes the activity to the laptop screen. Figure F-34. To stop the test at any time, press any key.

**FIGURE F-34. VBD Pump and Hold at Pressure Test**

```
20661.682,HVBD,N,Hit any key to abort!
20661.777,HVBD,N, cycle sec  vbd0  vbd1 avg mA P   psi
20662.137,HVBD,N,1  Pressure start: 0  end: 1483  VBD
          start: 3803  end: 3803
                  Time: 0  cc/sec: 0.00  avg mA: 0.000
20662.343,HVBD,N,----- pumped to 3833 3772 3802 (0)----
20662.483,SPOWER,N,powerOFF (3,VBD_pump_during_surface)
          without corresponding powerON!|
```

At the conclusion of the test, Seaglider returns to the VBD menu.

## *Special Test #1*

This test is done in a pressure chamber. It is not recommended that this test be done outside of the iRobot factory.

To test the motor current, ambient pressure, and potentiometers:

- Select **12**: Special Test #1

Seaglider returns:

```
Sample interval (ms)? [1000] |
```

If the default value is okay press ENTER. Otherwise type in a new value and press ENTER.

```
Max Pressure? [1500] 1|
```

Set the minimum pressure. For this example, pressure is set to .1

```
Max Pressure? [1500] 500|
```

Set the maximum pressure. For this example, pressure is set to .1

```
Aborts when pressure > 500 psia, or any key hit !
Hit CR to start|
```

Press ENTER to start the test

Seaglider starts the test and echoes the activity to the laptop screen. Figure F-35. To stop the test at any time, press any key.

**FIGURE F-35. Special Test #1**

```
ime(sec)   pumpCur linPot1 linPot2    linAvg      AmbP      psia
   0.621   554.541    3834    3769      3801    377263       0.3
   2.121   489.346    3821    3755      3788    377214       0.3|
```

At the conclusion of the test, Seaglider returns to the VBD menu.

## *Supervisor Menu*

To access the Supervisor menu:

- Select **4**: Supervisor and press ENTER

The menu options in Figure F-36 below are listed. This menu is used during the manufacture of Seaglider. It is not recommended that items in this menu be accessed outside of the iRobot factory.

**FIGURE F-36. Supervisor Menu**

```
        ------ Supervisor menu ------
      1 [sample  ]  Sample A-D channel
      2 [readad  ]  Read A-D register
      3 [monitor ]  Dump monitored A-D values
      4 [command ]  Send command
      5 [show    ]  Show gain, fuel, monitoring, heart and watchdog
        rates
      6 [setup   ]  Setup gain, fuel, monitoring, heart and watchdog
        rates
      7 [readcfg ]  Read configuration registers
      8 [accum   ]  Read accumulators
      9 [clear   ]  Clear accumulators
     10 [minmax  ]  Clear min/max values
     11 [write   ]  Write register value
     12 [read    ]  Read register value
     13 [setrtc  ]  Set RTC clock (from TT8 clock)
     14 [testrtc ]  Set setting RTC clock (from TT8 clock)
     15 [rtctott8]  Set TT8 clock from RTC
     16 [readrtc ]  Read RTC clock
     17 [status  ]  Read status bytes
     18 [zerochk ]  Check current zero points
     19 [zeroset ]  Set current zero points
     CR)  Return to previous
  Enter selection (1-19,CR): |
```

- To exit the Supervisor menu and return to the Hardware Menu, press ENTER.

## *Pressure Sensor Menu*

To access the Pressure Sensor menu:

- Select **5**: Pressure Sensor and press ENTER

The Pressure menu options appear (Figure F-37).

**FIGURE F-37. Pressure Sensor Menu**

```
------ Pressure menu ------
     1 [selftest]  Basic self-test
     2 [sealevel]  Sealevel test
     3 [param   ]  Edit pressure parameters
    CR)  Return to previous
Enter selection (1-3,CR):
```

The Basic self test outputs pressure measurements and an equivalent depth (m) value until a Crtl-Q is received.

In the Sealevel test, Seaglider takes a number (user determined) of pressure samples and calculates what the pressure y-intercept should be. The user has the option of accepting or rejecting the new y-intercept value.

The user can also edit the pressure parameters, although this is not recommended. iRobot Customer Service should be consulted before any pressure parameter values are changed via option 3.

Press ENTER to return to the Hardware Menu.

## *Compass/Attitude Menu*

To access the Compass/attitude Menu:

- Select **6**: Compass and press ENTER

The menu is Figure F-38 is displayed. This menu is used to set up the compass at the factory. It is not recommended that the user change any of the values in this menu or attempt a recalibration without consulting iRobot Customer Service.

**FIGURE F-38. Compass Menu**

```
------ Compass/attitude Menu ------
     1 [selftest]  Basic self-test
     2 [active  ]  Toggle active compass
     3 [dispraw ]  Display raw bearing, pitch & roll
     4 [dispcal ]  Display calibrated bearing, pitch & roll
     5 [direct  ]  Direct comms with unit
     6 [capture ]  Capture compass serial output
     7 [command ]  Send command to SP3003
     8 [reset   ]  Reset SP3003
     9 [whirly  ]  Whirly calibration - display raw sensor outputs
    10 [whirlraw]  Whirly calibration - (SP3003 only) display
       transducer counts
    11 [config  ]  Configure TCM2
    12 [calib   ]  Calibrate compass
    13 [calibraw]  Calibrate compass (w/ SP3003 raw raw reader)
    14 [auto    ]  Autocalibrate compass manual (cal stand) mode
    15 [insitu  ]  Autocalibrate compass in-situ mode
    16 [coeff   ]  Read calibration coefficients
    17 [power   ]  Power cycle test
    18 [edit    ]  Edit compass parameters
    CR)  Return to previous
Enter selection (1-18,CR):
```

- To exit the Compass/attitude menu and return to the Hardware Menu, press ENTER.

## *GPS Menu*

To view the GPS Menu:

- Select **7**: GPS and press ENTER

The menu is Figure F-39 is displayed. The menu options allow the user to check the basic functionality of the GPS unit.

**FIGURE F-39. GPS Menu**

```
------ GPS Menu ------
     1 [selftest]  GPS self-test (acquire fix)
     2 [display ]  Display lat/long & sat info
     3 [direct  ]  Direct comm with unit
     4 [capture ]  Capture raw output
     5 [reset   ]  Reset to deployment mode
     6 [check   ]  Check almanac status
     7 [pps     ]  Check for presence of PPS signal
     8 [clock   ]  Use GPS to set TT8 RTC (uses PPS if available)
     9 [version ]  Report GPS version
    CR)  Return to previous
```

- To exit the GPS Menu and return to the Hardware Menu, press ENTER.

## *Modem Menu*

To view the Iridium phone menu:

- Select **8**: Modem and press ENTER

The menu is Figure F-40 is displayed. The user can access this menu to check the basic functionality of the Iridium phone.

**FIGURE F-40. Iridium Phone Menu**

```
------ Iridium phone menu ------
     1 [selftest]  Self-test (does not make phone call)
     2 [direct  ]  Direct comms
     3 [signal  ]  Monitor signal quality
     4 [locate  ]  Geolocate phone
     5 [remote  ]  Remote login and PicoDOS commands (xr and xs enabled
                   for transfer)
     6 [upload  ]  Exercise uploadData
     7 [uploadst]  Exercise uploadData self-test-results only
     8 [current ]  Measure modem current
    CR)  Return to previous
Enter selection (1-8,CR): |
```

Press ENTER to return to the Hardware Menu.

## *Internal Pressure Menu*

To view the Internal pressure menu:

- Select **9**: Internal Pressure and press ENTER

The menu in Figure F-41 is displayed. The self test outputs internal pressure and relative humidity measurements until a Crtl-Q is received.

The internal pressure parameters are set at the factory. It is not recommended that the user edit these values without consulting iRobot Customer Service.

**FIGURE F-41. Internal Pressure Menu**

```
------ Internal pressure menu ------
     1 [selftest]  Basic self-test
     2 [param   ]  Edit internal pressure parameters
    CR)  Return to previous|
```

- To exit the Internal Pressure menu and return to the Hardware Menu, press ENTER.

## *Altimeter/XPDR Menu*

To view the Altimeter/XPDR Menu:

- Select **10**: Altimeter/XPDR Menu and press ENTER

The menu in Figure F-42 is displayed. Both the altimeter and the transponder functions can be queried from this menu. The unit is configured at the factory. It is not recommended that the user edit the configuration values without consulting iRobot Customer Service.

Adjusting the altimeter parameters is done in the field as sensitivity varies with the environment Seaglider is in.

**FIGURE F-42. Altimeter/XPDR Menu**

```
------ Altimeter/XPDR Menu ------
     1 [ping    ]  Ping the altimeter
     2 [config  ]  Upload configuration to altimeter
     3 [count   ]  Query the transponder ping count
     4 [xpdrcfg ]  Update XPDR configuration from parameters
     5 [direct  ]  Direct comms with altimeter
     6 [param   ]  Edit altimetry parameters
    CR)  Return to previous
```

- To exit the Altimeter Menu and return to the Hardware Menu, press ENTER.

## *Sensors Menu*

To view the Sensors menu:

- Select **11**: Sensors menu and press ENTER

A menu showing all of the installed science sensors is displayed. In the example below (Figure F-43), the SBE-CT sensors are the only ones installed.

**FIGURE F-43. Sensors Menu**

```
------ Sensors menu ------
           1 [sbect   ]  SBE_CT
          CR)  Return to previous
```

When the user selects a sensor, he is asked how many samples the sensor should take. The data is then displayed on the screen. If there are zeros in the counts

column, this means that there is no signal and the counter has timed out. The user needs to understand the reason for the lack of data and resolve the issue.

- To exit the Sensors menu and return to the Hardware Menu, press ENTER.

## *Loggers Menu*

To view the Sensors menu:

- Select **12**: Loggers and press ENTER

A menu showing all of the installed science sensors is displayed. In the example below (Figure F-44), the GPCTD is the only logger device installed.

**FIGURE F-44. Example Menu**

```
------ Loggers menu ------
     1 [pc      ]  GPCTD
     2 [        ]  empty
    CR)  Return to previous
Enter selection (1-2,CR): 1
```

When the user selects a logger device a listing of all the functions available for that device will appear (Figure F-45).

**FIGURE F-45. Loggers Menu**

```
------ Logger device test menu ------
     1 [on      ]  Turn on
     2 [off     ]  Turn off
     3 [selftest]  Selftest
     4 [sample  ]  Report a sample
     5 [syncclk ]  Synchronize device clock to TT8
     6 [clock   ]  Read device clock
     7 [file    ]  Get file from device
     8 [action  ]  Execute logger action
     9 [config  ]  Show configuration
    10 [edit    ]  Edit configuration
    11 [direct  ]  Direct comms
    CR)  Return to previous
```

## *Batteries and Fuel Gauges Menu*

To view the Batteries and Fuel Gauges menu:

- Select **13**: Batteries and Fuel Gauges Menu and press ENTER

The menu in Figure F-46 is displayed. When viewing the battery gauges (selection 1) the third column (since power-up) should contain zeros if the Seaglider is running on external power. However, if Seaglider is running on internal power the third column should be populated with non-zero values.

Resetting the battery gauges (selection 2) is not recommended. This is a record of the usage and remaining life of the installed batteries.

Backup battery gauges (selection 3) are not installed in iRobot gliders.

The battery voltages option (selection 4) lists the present voltage on the 24V and 10V battery packs.

**FIGURE F-46. Batteries and Fuel Gauges Menu**

```
    ------ Battery Functions ------
 1 [read    ]  View battery gauges
 2 [reset   ]  Reset battery gauges
 3 [backup  ]  Backup battery gauges
 4 [voltage ]  Battery voltage
CR)  Return to previous
```

- To exit the Batteries and Fuel Gauges menu and return to the Hardware Menu, press ENTER.

## *Low-Level Hardware Menu*

To view the Low-level hardware functions menu:

- Select **14**: Low-level Hardware and press ENTER

The menu is Figure F-47 is displayed. This menu is used during manufacture and if there is an apparent low level hardware failure. Most users will never need to access this menu. If access is deemed necessary, it should be done at the direction of iRobot Customer Service.

**FIGURE F-47. Low-level Hardware Menu**

```
------ Lowlevel hardware functions ------
*I/O lines
     1 [io       ]  Set and read IO lines
     2 [watchdog]  Hang and test the watchdog
     3 [stroke   ]  Repetitively stroke the watchdog
*A/D converters
     4 [int      ]  Internal A/D
*Power control
     5 [measure ]  Measure device power
     6 [report   ]  Report active devices
*CompactFlash
     7 [cf       ]  CF card evaluation
     8 [size     ]  Report CF card size
*TPU
     9 [terminal]  Serial port terminal mode
    CR)  Return to previous
Enter selection (1-10,CR):
```

- To exit the Low-level hardware functions menu and return to the Hardware Menu, press ENTER.

## *Miscellaneous Hardware Functions Menu*

To view the Miscellaneous hardware functions menu:

- Select **15**: Miscellaneous Menu and press ENTER

The menu in Figure F-48 is displayed. With the exception of the first (Prepare for Travel) and fourth (Read/set time-of-day) menu items, it is not recommended that the user access or change any of the other menu functions without consulting iRobot Customer Service.

**FIGURE F-48. Miscellaneous Menu**

```
   ------ Miscellaneous hardware functions ------
 1 [travel  ]  Prepare for travel
 2 [iotime  ]  Change user IO timeout
 3 [timeout ]  Change watchdog time (mins)
 4 [date    ]  Read/set time-of-day
 5 [zerost  ]  Reset self-test counter to zero
 6 [fault   ]  Force a fault - WARNING, CRASHES to TOM8!!!!
 7 [dump    ]  Dump captured registers
 8 [walk    ]  Heap walk
CR)  Return to previous
Enter selection (1-5,CR): |
```

When the user selects 1: Prepare for travel, Seaglider automatically puts itself into travel mode. It bleeds the external bladder, pitches fully forward, rolls to neutral and finishes the process with the statement “You are good to go!”. Each step of this process is echoed to the laptop screen.

When the user selects 4: Read/set time-of-day, the same procedure is used here as it is when Seaglider is first turned on. See step 11 on page 192.

- To exit the Miscellaneous menu and return to the Hardware Menu, press ENTER.

## *Developer Tests Menu*

To view the Developer Tests menu:

- Select **16**: Developer Tests menu and press ENTER

The menu in Figure F-49 is displayed. This menu is for Seaglider developers and should not be used by customers.

**FIGURE F-49. Developer Tests Menu**

```
------ Developer functions ------
     1 [lpsleep ]  Test low-power sleep
     2 [report  ]  Report memory status
     3 [irq     ]  Test interrupt contention
     4 [toggle  ]  Toggle clearscreen
     5 [file    ]  Test file creation/manipulation
    CR)  Return to previous
Enter selection (1-5,CR):
```

- To exit the Developer Tests menu and return to the Hardware Menu, press ENTER.
- To exit the Hardware Menu and return to the Main Menu, press ENTER again.

## *Test Operation Modes amd Files*

To view the Test Operation Modes and Files Menu:

- Select **3**: Test Operation Modes and Files and press ENTER

The menu in Figure F-50 is displayed. Selection 1:Test Bathymetry Files is the check that users will most often perform from this menu. It tests the successful loading of bathymetry files onto Seaglider. This is the same check that is done during a self test.

Selections 2 – 8 are used during Seaglider checkout at the factory and at the direction of iRobot Customer Service if a problem occurs with Seaglider.

**FIGURE F-50. Test Operation Modes and Files**

```
------ Operational Functions Test Menu ------
     1 [bathy   ]  Test bathymetry files
     2 [recovery]  Test recovery
     3 [surface ]  Test surface maneuver
     4 [sample  ]  Test sampling and data file creation
     5 [angle   ]  Test surface measurements normal
     6 [upload  ]  Exercise uploadData regular operations
     7 [uploadst]  Exercise uploadData self-test results only
     8 [modes | ]  Test active/passive modes
    CR)  Return to previous
Enter selection (1-8,CR):
```

- To exit the Hardware Menu and return to the Main Menu, press ENTER again.

## *PicoDOS Commands*

To view the picoDOS commands menu:

- Select 4: PicoDOS Commands (and exit) and press ENTER

Seaglider responds with:

```
Enter extended and normal picoDOS commands at the prompt
 quit: returns to menu
 pdos: exits the program to pDOS
 tom8: exits the program to TOM8
??: gives help and hints
```

For more information on picoDOS commands see Appendix C: Extended PicoDOS Reference Manual.

It is not recommended that the user exit down to the pdos or TT8 levels.

## *Pre Launch Menu*

To view the Pre Launch Menu, Figure F-51:

- Select **5** and press ENTER

**FIGURE F-51. Pre Launch Menu**

```
------ Launch Menu ------
     1 [scene   ]  Set scenario mode
     2 [selftest]  Perform interactive self test
     3 [autotest]  Perform autonomous self test
     4 [uploadst]  Upload self-test results
     5 [reset   ]  Reset dive/run number
     6 [test    ]  Test Launch!
     7 [sea     ]  Sea Launch!
    CR)  Return to previous
```

## *Set Scenario Mode*

To access the Set Scenario Mode menu:

- Select **1**: Set Scenario Mode and press ENTER

Scenario mode is used with Seaglider in a test tank. This mode allows the user to program Seaglider to go through a series of pitch, roll and/or VBD maneuvers. The user determines the amount of motor or pump movement made during each step and the length of the scenario.

Follow the protocols shown with each step when designing the scenario.

At the end of the scenario, Seaglider returns to the Scenario menu.

- To exit the Scenario menu and return to the Pre Launch Menu, press ENTER .

## *Perform Interactive Self Test*

To access the Interactive Self Test Mode menu:

- Select **2**: Perform Interactive Self Test and press ENTER

Follow the interactive self test directions in Chapter 7, “Pre-Launch Procedures” on page 187.

At the end of the interactive self test, Seaglider returns to the Launch menu.

## *Perform Autonomous Self Test*

To access the Autonomous Self Test Mode menu:

- Select **3**: Perform Autonomous Self Test and press ENTER

Follow the autonomous self test directions in Appendix E, “Autonomous Self Test” on page 347.

At the end of the autonomous self test, Seaglider returns to the Launch menu.

## *Upload Self Test Results*

To upload self test results to the laptop connected to Seaglider via the serial communications cable:

- Select **4**: Upload Self Test Results and press ENTER

The output looks like that in Figure 7-30 on page 213.

At the end of the upload that Seaglider returns to the Launch menu.

## *Reset Dive/Run Number*

To reset the dive or run number:

- Select **5**: Reset Dive/Run Number and press ENTER

Seaglider responds with the present dive number. If this value is correct press ENTER. Otherwise, type in a new dive number and press ENTER.

Seaglider returns to the Launch menu.

## *Test Launch*

To run simulated dives with Seaglider:

- Select **6**: Test Launch! and press ENTER

Follow the simulated dives procedure in Chapter 7, “Pre-Launch Procedures” on page 187.

At the end of the test Seaglider returns to the Launch menu.

## *Sea Launch*

To run the last launch sequence before deploying a Seaglider:

- Select **7**: Sea Launch! and press ENTER

Follow the sea launch procedure in Chapter 6, “Pre-Deployment Tasks” on page 159.

## *Serial Port Data Transmission*

Directly connect to the Seaglider via the communications cable and launch a terminal program (TeraTerm preferred)

1. Press enter to return to the main menu.
2. From the main menu select pdos (4)
3. Enter the command to zip up all the files on the CF card into a .tar file

   picoDOS>tar c temp.tar *
4. Enter into extended Picodos

   picoDOS>> pdos
5. When asked “Really exit to pdos? [N]” type y
6. Change the baud rate to 38400

   PicoDOS>baud 38400
7. Change TeraTerm’s baud rate to 38400 using the drop down menu.

   Setup > Serial Port…
8. Start the xmodem transfer on the Seaglider

   PicoDOS>xs temp.tar
9. Start the xmodem transfer on TermTerm using the drop down menu

   File>Transfer>xmodem>Receive
10. Move the .tar file from your laptop to the Seagliders directory (sgxxx) on the basestation with file transfer software (WinSCP...)
11. Navigate to the Seagliders directory on the basestation with a terminal emulator (Putty...)
12. Unzip the .tar file (where "tar file name = the name of the .tar file) [xxxx@base sgxxx]$ tar xf Temp."tar file name"
13. Enter the touch command

    [xxxx@base sgxxx]$ touch comm.log
14. Enter the script to change the file extension [xxxx@base sgxxx]$ for file in `ls *.A`; do newfile=${file:0:8}; newfile=`echo $newfile | awk '{print tolower($0)}'`.x00; cp $file $newfile; done
15. Force the basestation data processing [xxxx@base sgxxx]$ python /usr/local/basestation-2.05/Base.py -m . --force --verbose

Your processed data will now be in the Seagliders directory on the basestation.

APPENDIX G

# *Ballasting*

This section describes ballast of your Seaglider.

## *Overview*

Seaglider is ballasted externally to the pressure hull. The ballast areas are the nose of the forward fairing, the circumference of the electronics hull, and the rearmost battery hull. Ballast for the Seaglider is typically located in the nose and rearmost battery hull. The area around the electronics hull is rarely used. The ballast medium in the nose of the fairing is brass plates. The rearmost ballast can be in the form of lead strips (100g – 180g respectively) and/or syntactic foam strips.

Proper ballasting of the Seaglider allows the vehicle to optimally perform in the operational environment. Initially, based upon customer input, the Seaglider is delivered to the customer ballasted for the approximate bottom water density and maximum stratification in the deployment area. The parameters needed by iRobot to complete the pre-delivery ballasting are:

1. estimated bottom water density in either $kg/m_3$ or $g/cm_3$
2. estimated surface water density in either $kg/m_3$ or $g/cm_3$
3. expected current in the deployment area (kts or m/s) or the desired thrust (cc's)

Based upon the customer provide information, iRobot will calculate and affix the appropriate weights to the outside of the pressure hull and provide the customer with the calculated C_VBD, C_PITCH, C_ROLL_DIVE, and C_ROLL_CLIMB for the mission environment. The pilot will fine tune these values at the beginning of the deployment.

## *Re-ballasting in the field*

Should the glider need to be re-ballasted for a different environment between deployments, contact iRobot for assistance. iRobot will provide new weight estimates and the positions on the glider where the weights should be changed.

Tools & Material Required for Re-ballasting

- #2 Phillips Screwdriver
- Flat Blade Screwdriver
- 4oz ball peen hammer
- Isopropyl Alcohol
- Lead Ballast Strips
- Syntactic Foam Strips
- Carpet Tape
- Scotch Super 88 Vinyl Electrical Tape (1.5in wide)
- Box Knife
- Pupa Straps (either large zip ties or plastic banding)
- Tef-Gel
- Brass Screws and Lock Washers for Nose Weight Plates
- 3/8in Ratchet, 36in 3/8in extension, 12in 3/8in extension 3/16in allen with 3/8” socket
- Nose weight installation tool
- Needle Nose Pliers

The nose weight plates can be added or removed by removing the two brass socket head cap screws securing the nose weight(s) to the nose of the forward fairing. If adding weight plates, apply a coating of Tef-Gel to the plate surfaces that will come in contact with one another before installing. The screws needed to attach the nose

plates are listed below and are determined by the number of nose weight plates used.

| Qty | Qty of 1/4in Nose Plates | Qty of 1/8in Nose Plates | Screw Length (1/4in – 20 ) |
|---|---|---|---|
| 2 | 1 | 2 | 3/4in |
| 2 | 2 to 3 | 4 to 6 | 1 1/2in |
| 2 | 4 to 5 | 8 to 10 | 2.0in |

Lead ballast on the exterior of the rear-most battery hull is affixed using carpet tape and held in place with the straps around the battery hull. The lead ballast is isolated from the hull using an EPDM rubber pad. The pad is also present to protect the pupa anodizing and allow for expansion and contraction of the hull.

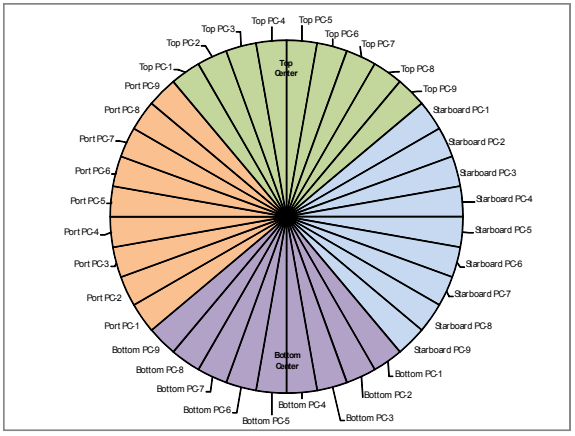
Steps to adjust lead ballast

1. Remove Forward Fairing.
2. Take note of lead values and location on the pupa. (Refer to the lead worksheet tab on Trim Sheet) When changing ballast by adding or subtracting lead, the changes must be made according to the diagram provided by iRobot.
3. Remove straps from rearmost battery hull. If zip ties are used as the strapping material, cut them off over the rubber area so as not to damage anodizing on the battery hull. If buckles and straps are used as the strapping material, release the buckle with a flat blade screwdriver.
4. If removing or moving lead strips, use a flat screwdriver to pry up one side of the lead from the tape and pull off.
5. Remove the old carpet tape, clean the area with isopropyl alcohol and a lint free cloth, to remove any remaining glue or dirt from the tape, and apply new tape.

6. Place new lead ballast on the pupa using the diagram supplied by iRobot. An example of the drawing showing positioning of the new lead is shown in Figure A-1.
7. Using the ball peen hammer lightly tap the lead until it conforms to the general shape of the battery hull.
8. Install 2 new zip ties or buckled straps. Make sure that the straps are pulling opposite one another and that the buckles are offset from one another. The further the buckle offset the better.
9. Record lead values and location on the battery hull. See ballast pin-wheel (i.e. 1st piece of lead trim is located in the bottom quadrant PC-1)
10. Re-install forward fairing

Note: If lead is moved between deployments to a different location on the pupa the pilots will have to trim roll during the first few dives.

**FIGURE A-1. Ballast Location Pin-wheel**




If syntactic foam, rather than lead, is needed to ballast the glider affix it to the Seaglider using carpet tape and secure in place using Scotch Super 88 Vinyl Tape 1.5in wide. The carpet tape is applied to the battery hull in a thin strip. The foam is stuck to the tape in pre-defined positions based on the diagram provided by iRobot.

The tape is applied partially on the foam and partially on the pupa to create a sealed pocket for the foam to sit in.

Steps to apply or adjust Syntactic foam ballast

1. Remove Forward Fairing.
2. Take note of foam values and location on the pupa. (Refer to the lead worksheet tab on Trim Sheet). When changing ballast by adding or subtracting foam, the changes must be made according to the diagram provided by iRobot.
3. Remove the installed wraps of vinyl tape making sure that none of the foam pieces are lost.
4. If the foam falls off, remove all the foam in order and then remove the strip of carpet tape.
5. Clean the pupa with isopropyl alcohol and dry with a lint free cloth. This will remove any remaining glue or dirt from the tape.
6. Install a new strip of carpet tape that is long enough to accommodate all the foam.
7. Reapply the required amount of syntactic foam to the carpet tape.
8. Reapply the vinyl tape keeping half the tape on the foam and half on the pupa. Make three complete wraps.
9. Record foam values and location on the battery hull. See ballast pin-wheel (i.e. 1$^{st}$ piece of trim is located in the bottom quadrant PC-1)
10. Reinstall forward fairing.

Note: If foam is moved, between deployments, to a different location on the pupa the pilots will have to trim roll during the first few dives.

APPENDIX H

# *Pilot and Field Team Checklist*

**Prepare the necessary software on field laptop.**

| Step | Responsibility | Action |
|---|---|---|
| 1. | Field Team | Terminal Program ☐ |
| 2 | | Text Editor ☐ |

**Prepare the necessary software on pilot computer.**

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot | Secure shell program ☐ |
| 2 | | Navigational plotting program ☐ |
| 3 | | MatLab ☐ |
| 4. | | Text Editor ☐ |

## Obtain Pilot and Field Team contact information.

| Step | Responsibility | Action |
|---|---|---|
| 1. | Field Team | Phone Number: |
| 2 | | E-mail address: |
| 3 | | Sattellite phone (if applicable): |

## Obtain Pilot contact information.

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot | Phone Number: |
| 2 | | E-mail address: |

## Obtain Basestation information

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot and Field Team | Telnum: |
| 2 | | Altnum: |
| 3 | Pilot | IP address: |
| 4 | | Username: |
| 5 | | Password: |

## Self-Test Schedule

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot and Field Team | Location: |
| 2 | | Date: |
| 3 | Field Team | Time self-test will be ready for review. |
| 4 | Pilot | Time self-test need to be completed. |

## Deployment Schedule

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot and Field Team | Location: |
| 2 | | Date: |
| 3 | | Mission planning details should be organized (refer to “Pre-Deployment Tasks” on page 159) : |
| 4 | Pilot | targets, science and command files generated : |
| 5 | Field Team | Time Field Team plans to be at deployment site: |
| 6 | | Deployment Vessel being used: |

## Recovery Schedule

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot and Field Team | Location: |
| 2 | | Date: |
| 3 | Field Team | Time Field Team plans to be at recovery site: |
| 4 | Pilot | Time and location Pilots will plan to have the Seaglider on the surface and ready for recovery: |
| 5 | Field Team | Recovery vessel being used: |

## Self-Test Process

| Step | Responsibility | Action | |
|---|---|---|---|
| 1. | Pilot | Logs into basestation and prepares to analyze Self-Test data files. | ☐ |
| 2 | Field Team | Performs a Self-Test on the Seaglider at the designated time. | ☐ |
| 3 | | Informs the Pilot when the Self-Test has completed. | ☐ |
| 4 | Pilot | Analyzes the Self-Test data and fills out the Self-Test log sheet as required. | ☐ |
| 5 | | Informs the Field Team if all systems are good and the Self-Test passes. | ☐ |
| 6 | | Saves the Self-Test log sheet onto the basestaton. | ☐ |
| 7. | | Gives the Filed Team approval to put the Seaglider into sea-launch mode. | ☐ |
| 8 | Pilot and Field Team | Can now begin to prepare for sea-launch. | ☐ |

## Deployment Process

| Step | Responsibility | Action |
|---|---|---|
| 1. | Pilot | Logs into the basestation via secure shell method. ☐ |
| 2. | Field Team | Arrives at designated sea-launch site. ☐ |
| 3. | | Ensures the antenna O-ring is present and fully tightened ☐ |
| 4. | | Ensures all cables and connectors are secure ☐ |
| 5. | | Assembles Seglider's wings, rudder and antenna ☐ |
| 6. | | Removes sensor covers ☐ |
| 7. | | Requests permission to sea-launch the Seaglider with a tether attached. ☐ |
| 8. | Pilot | Approves the sea-launch. ☐ |
| 9. | Field Team | Launches the Seaglider with tether attached. ☐ |
| 10. | | Acoustically pings the Seaglider with a transducer and deckbox. ☐ |
| 11. | | Reports back to Pilot the buoyancy quality of the Seaglider by describing how the Seaglider is sitting in the sea surface. ☐ |
| 12. | Pilot | Prepares the Seaglider to dive if buoyancy is correct and all systems are good. ☐ |
| 13. | | Instructs the Field Team to release the tether. ☐ |
| 14. | Field Team | Reports to the Pilot once the Seaglider has submerged and is diving. ☐ |
| 15. | | Remains onsite while the Pilot analyzes the dive data. ☐ |
| 16. | Pilot | Insures the Seaglider is operating correctly. ☐ |
| 17. | | Informs the Filed Team they are clear to leave the sea-launch site area. ☐ |

## Flight Process

| Step | Responsibility | Action | |
|---|---|---|---|
| 1. | Pilot | Transfers dive data from the basestation onto their piloting computer for each dive. | ☐ |
| 2. | | Plots the dive data with MatLab software. | ☐ |
| 3. | | Trims the Seaglider as needed by analyzing the MatLab plots and making the required parameter changes. | ☐ |
| 4. | | Looks for any odd flight behavior. | ☐ |
| 5. | | Makes sure the science data is present and each sensor is producing good quality data. | ☐ |
| 6. | | Makes sure the altimeter can find the bottom and is correctly tuned. | ☐ |
| 7. | | Fills out dive log sheet if needed. | ☐ |
| 8. | | Plots the Seagliders current GPS coordinates with navigational plotting software during each surfacing to monitor its current location. | ☐ |
| 9. | | Monitors the tide and ocean current information to assist with predicting waypoint targeting. | ☐ |
| 10. | | Flies the Seaglider as required between designated waypoints. | ☐ |

## Recovery Process

| Step | Responsibility | Action | |
|---|---|---|---|
| 1. | Pilot | Ensures the Seaglider is flying towards the recovery waypoint in advance. | ☐ |
| 2. | Field Team | Notifies Pilot once they leave port on recovery vessel and their estimated steam-time to be onsite. | ☐ |
| 3. | Pilot | Changes the necessary parameters to put the Seaglider into shallow dives. | ☐ |
| 4. | Field Team | Arrives at designated recovery location at designated time and informs the Pilot they're onsite. | ☐ |
| 5. | Pilot | Ensures the Seaglider arrives at the designated recovery location at the designated time. | ☐ |
| 6. | | Puts the Seaglider into recovery mode and insures it will remain on the surface. | ☐ |
| 7. | | Notifies Field Team once the Seaglider is on the surface and in recovery mode. | ☐ |
| 8. | | Relays the last known GPS coordinates to the Field Team. | ☐ |
| 9. | Field Team | Searches for the Seaglider on the sea surface. | ☐ |
| 10. | | Recovers the Seaglider. | ☐ |
| 11. | | Informs the Pilot the Seaglider has been successfully recovered | ☐ |
| 12. | | Attaches the necessary sensor protective covers. | ☐ |
| 13. | | Poweres the Seaglider off | ☐ |
| 14. | | Makes sure the Seaglider is powered off | ☐ |
| 15. | Pilot | Completes the dive log sheet and saves it onto the basestation if needed. | ☐ |

| | | | |
|---|---|---|---|
| 16. | Field Team | Steams back to port. | ☐ |
| 17. | | Properly cleans the Seaglider and all sensors. | ☐ |
| 18. | | Puts the Seaglider into travel mode and properly stows it away. | ☐ |

APPENDIX I

# *Sample Field Kit Checklist*

## *Field Kit Checklist*

| Check complete | Field Kit Checklist Task |
|---|---|
| | 1KA Seaglider<br>(Self Tested with known Transducer Frequency and other parameters) |
| | Seaglider Cal Sheets - MOST CURRENT VERSION |
| | Laptop computer (charged overnight) with power adapter |
| | Field phone with power adapter (field location determines phone type - cell, Iridium) |
| | Handheld GPS unit with extra batteries |
| | Benthos DS-7000 (charged overnight) |
| | Benthos transducer safety line/ Seaglider tag line/cradle safety line |
| | Marine radio (check batteries) |
| | Power inverter (w/ spare fuses) and 12V battery (charged) |
| | Extension cord |
| | Plug strip |
| | (2x) Communication cable (50 feet) |

| Check complete | Field Kit Checklist Task |
|---|---|
| | Communication cable (10 feet) |
| | Sensor rinse water bottle |
| | Gallon of distilled water |
| | (2x) Magnetic wand |
| | Notebook and pencils or pens |
| | Extra screws (rudder, fairing) |
| | T-Handle (hex) driver for rudder screws |
| | Phillips screwdriver for fairing screws |
| | (3x) Quick clamps |
| | Foul weather gear |
| | Life preservers |
| | Binoculars |
| | Boots or closed-toe shoes |
| | Tide chart for date of launch/recovery |
| | Sensor plugs/covers |
| | Rinse hose |
| | Paper towels |
| | Tie down straps and bungee cords |
| | Sunscreen |
| | Drinking water |
| | Sunglasses |
| | |
| | |

APPENDIX J

# *Warranty and Disclaimers*

## *Seaglider™ Warranty and Disclaimers*

### 1. Warranty, Exclusive Remedies and Warranty Disclaimers.

1.1 **Warranty**. iRobot WARRANTS THAT DEVICES SOLD HEREUNDER SHALL BE FREE FROM DEFECTS IN MATERIALS AND WORKMANSHIP UNDER NORMAL USE AND SERVICE WHEN CORRECTLY OPERATED IN ACCORDANCE WITH THE iRobot INSTRUCTIONS AND TRAINING FOR A PERIOD OF ONE (1) YEAR FROM DATE OF CUSTOMER ACCEPTANCE. (THE "WARRANTY"). iRobot RESERVES THE RIGHT AND SOLE DISCRETION TO MODIFY THIS WARRANTY AT ANY TIME WITH WRITTEN NOTICE. CUSTOMER'S RECEIPT OF ANY DEVICE DELIVERED HEREUNDER SHALL BE AN UNQUALIFIED ACCEPTANCE OF AND A WAIVER BY CUSTOMER OF THE CUSTOMER'S RIGHT TO MAKE A CLAIM WITH RESPECT TO SUCH DEVICE UNLESS CUSTOMER GIVES iRobot WRITTEN NOTICE OF ANY CLAIM WITHIN ONE YEAR AFTER THE RECEIPT OF SUCH DEVICE.

1.2 **Exclusive Remedy**. The sole obligation of iRobot, and Customer's sole and exclusive remedy for a breach of the warranty in Section 1, shall be that iRobot shall use commercially reasonable efforts to repair and correct, or, at iRobot's option, replace the Device which shall have been promptly reported in writing as

not operating in accordance with the Warranty and, upon inspection by iRobot shall be determined to not have met the Warranty, provided the Device was not abused or operated other than in accordance with the iRobot instructions. If iRobot is unable to repair or correct the Software in a reasonable amount of time in accordance with this Section 1, iRobot will refund to Customer the amount paid for such Device, in which case Customer shall return the Device to iRobot.

1.3 **Assignment**. iRobot HEREBY ASSIGNS TO CUSTOMER ALL OF THE MANUFACTURERS' WARRANTIES RELATING TO THE EQUIPMENT WHICH iRobot IS PERMITTED BY THE EQUIPMENT MANUFACTURER(S) TO ASSIGN TO CUSTOMER. SUCH ASSIGNMENT IS SUBJECT TO ALL OF THE TERMS AND CONDITIONS IMPOSED BY THE EQUIPMENT MANUFACTURER(S) WITH RESPECT THERETO. iRobot WILL USE COMMERCIALLY REASONABLE EFFORTS TO PROMPTLY APPRISE CUSTOMER OF SUCH WARRANTIES FOLLOWING CUSTOMER'S PURCHASE OF THE EQUIPMENT.

1.4 **Disclaimers**. EXCEPT FOR THE FOREGOING EXPRESS WARRANTY SPECIFIED ABOVE, iRobot GRANTS NO WARRANTIES, EITHER EXPRESS OR IMPLIED. iRobot EXPRESSLY DISCLAIMS THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE, AND NONINFRINGEMENT FOR THE DEVICE(S), IN WHOLE AND IN PART. iRobot DOES NOT ASSUME LIABILITY FOR LOSS, DAMAGE, OR OTHER RESULTS OF OPERATING THE DEVICE IN WATER, OR DUE TO LEAKAGE, IMPLOSION, OR EXPLOSION. The warranty will not apply to the Device if there is a failure of the Device or any part thereof which is attributable to: (a) inappropriate or unauthorized use of the Device; (b) accident, neglect, misuse or abuse of the Device; (c) exposure of the Device to potentially harmful environmental, power, and operating conditions; (d) customer specific modifications not performed by iRobot without receipt of applicable training. iRobot does not warrant: that the functions contained in the Device will meet Customer's requirements; that the operation of the Device will be uninterrupted or error-free; or that all defects will be corrected.

### 2. Limitations of Liability and Disclaimer of Damages.

2. 1 LIABILITY FOR USE OR LOSS AT SEA. CUSTOMER SHALL BE ENTIRELY RESPONSIBLE FOR ANY LIABILITY RESULTING FROM USE OR LOSS OF DEVICE AT SEA, INCLUDING REGULATIONS AND RESTRICTIONS OF THE MARINE POLLUTION ACT (MARPOL), THE ENVIRONMENTAL PROTECTION AGENCY, THE UNITED STATES COAST

GUARD, ANY ACT THAT ENABLES THE EXISTENCE OF MARINE PROTECTED AREAS (BOTH U.S. AND OTHER), AND ANY OTHER STATE, REGIONAL, COUNTY, OR LOCAL REGULATIONS.

2.2 DISCLAIMER OF DAMAGES. NOTWITHSTANDING ANYTHING TO THE CONTRARY IN THIS AGREEMENT, IT IS EXPRESSLY AGREED THAT iRobot AND ITS SUPPLIERS, SHALL IN NO EVENT BE LIABLE FOR TORTIOUS CONDUCT (INCLUDING BUT NOT LIMITED TO NEGLIGENCE OR STRICT LIABILITY) OR INDIRECT, INCIDENTAL, SPECIAL OR CONSEQUENTIAL DAMAGES RELATING TO OR ARISING OUT OF THE AGREEMENT, EVEN IF iRobot IS ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. SUCH EXCLUDED DAMAGES INCLUDE, BUT ARE NOT LIMITED TO, LOSS OF GOODWILL, INTERRUPTION OF BUSINESS, DEVICES NOT BEING AVAILABLE FOR USE, LOST OR CORRUPTED DATA, LOSS OF BUSINESS, LOSS OF PROFITS, LOSS OF USE OF THE DEVICE OR ANY ASSOCIATED EQUIPMENT, COST OF CAPITAL, COST OF SUBSTITUTE OR REPLACEMENT PRODUCT, FACILITIES OR SERVICES, DOWN-TIME, CHARGES FOR CUSTOMER'S TIME AND EFFORT, THE CLAIMS OF THIRD PARTIES, INJURY TO PROPERTY, OR ANY OTHER DIRECT, INDIRECT, SPECIAL, RELIANCE, INCIDENTAL OR CONSEQUENTIAL DAMAGES, REGARDLESS OF THE NATURE OF THE CLAIM AND WHETHER OR NOT FORESEEABLE, AND WHETHER OR NOT BASED ON BREACH OF WARRANTY, CONTRACT OR TORT OR STRICT LIABILITY, OR FOR ANY CLAIM BY ANY THIRD PARTY EXCEPT AS EXPRESSLY PROVIDED HEREIN.

2.3 LIMITATION OF LIABILITY. ANY DAMAGES THAT iRobot SHALL BE REQUIRED TO PAY SHALL BE LIMITED TO THE TOTAL FEES AND CHARGES RECEIVED FROM CUSTOMER UNDER THIS AGREEMENT.

2.4 NO LIABILITY FOR DELAYS. iRobot AND ITS SUPPLIERS SHALL NOT BE LIABLE FOR ANY DAMAGES CAUSED BY DELAY IN DELIVERY, INSTALLATION OR OPERATION OF THE SOFTWARE OR EQUIPMENT UNDER THE AGREEMENT.